SB-36610UX SYNCHRO / RESOLVER USB TEST SYSTEM HARDWARE/SOFTWARE MANUAL

MN-36610UX-001

105 Wilbur Place
Bohemia, New York 11716-2426
Tel: (631) 567-5600, Fax: (631) 567-7358
World Wide Web - http://www.ddc-web.com

For Technical Support - 1-800-DDC-5757 ext. 7771
United Kingdom - Tel: +44-(0)1635-811140, Fax: +44-(0)1635-32264
France - Tel: +33-(0)1-41-16-3424, Fax: +33-(0)1-41-16-3425
Germany Tel: +49-(0)89-15 00 12-11, Fax: +49-(0)89-15 00 12-22
Japan - Tel: +81-(0)3-3814-7688, Fax: +81-(0)3-3814-7689
Asia - Tel: +65 6489 4801

Revision	Date	Pages	Description
Rev A	Dec. 2010	All	Initial release
Rev B	March 2011	Various	Added separate driver installation instructions for Windows Vista and 7. Added optional mounting installation instructions. Updated product picture. Clarified BIT output and accuracy specification. Added accessories to accessory list.
Rev C	October 2015	Various	Added Section 5.5 and 5.6, added 2-speed and mating connector information
Rev D	July 2016	78	Updated wiring instructions
Rev E	April 2019	79	Added frequency range for reference oscillator output

1 PREFACE

This manual uses typographical conventions to assist the reader in understanding the content. This section will define the text formatting used in the rest of the manual

1.1 Text Usage

- **BOLD**–indicates important information and table, figure, and chapter references.
- ***BOLD ITALIC***–designates DDC Part Numbers.
- `Courier New`–indicates code examples.
- <…> - indicates user-entered text or commands.

1.2 Standard Definitions

USB Universal Serial Bus

1.3 Trademarks

All trademarks are the property of their respective owners.

1.4 What is included in this manual?

This manual contains a complete description of hardware/software installation and use.

1.5 Supporting Documentation

- RD/RDC Series Converters Applications Manual (***MN-19220XX-001***)
- Synchro/Resolver Conversion Handbook
- Two-Speed Application Note (***AN/MFT-10***)
- ***RD-19230*** Series Data Sheet

1.6 Technical Support

In the event that problems arise beyond the scope of this manual, you can contact DDC by the following:

US Toll Free Technical Support:
1-800-DDC-5757, ext. 7771

Outside of the US Technical Support:
1-631-567-5600, ext. 7771

Fax:
1-631-567-5758 to the attention of Motion Feedback Techonologies Applications

DDC Website:
www.ddc-web.com/ContactUs/TechSupport.aspx

Please note that the latest revisions of Software and Documentation are available for download at DDC's Web Site, www.ddc-web.com.

2 OVERVIEW

The ***SB-36610UX*** is a USB interface device which contains two channels of fully independent Synchro/Resolver-to-Digital conversion. For each channel the conversion process is implemented using a DDC ***RD-19230*** 16-bit monolithic converter. This form factor is designed for use on any system containing a USB port such as desktops, laptops and netbooks.

2.1 Features

- USB interface
- Two independent input channels
  - Input amplitudes: 0.34 Vrms to 90 Vrms
  - 47 Hz to 10 kHz operating carrier frequency
  - 1 Arc minute accuracy
  - 10, 12, 14, 16 bit resolution
  - 15/45 Hz or 80/300 Hz bandwidth options
  - Software programmable resolution and bandwidth
  - A Quad B incremental encoder outputs (A, B, ZI)
  - Analog and Digital velocity outputs
  - Synthesized reference
  - Built-in-test (BIT) output
- On-board programmable reference oscillator
  - Output voltage: 2 – 32 Vrms and 20 – 123 Vrms options
  - 400 Hz to 10 kHz operating frequency
  - 2 VA output drive
  - Software programmable voltage and frequency
- Built-In self test
- Lead-free
- Included Software
  - Complete Synchro C SDK
    - Plug-n-Play Windows® XP/Vista/7 /8(32/64-bit) drivers
    - Abstracts all low-level hardware memory/registers
  - Windows Graphical User Interface (GUI)
    - User-friendly application that demonstrates full capabilities of the device

2.2 System Requirements

- USB 1.1 / 2.0 (Full-Speed) Port
- Windows® XP/Vista/7/8

Figure 1. SB-36610UX Synchro / Resolver USB Device

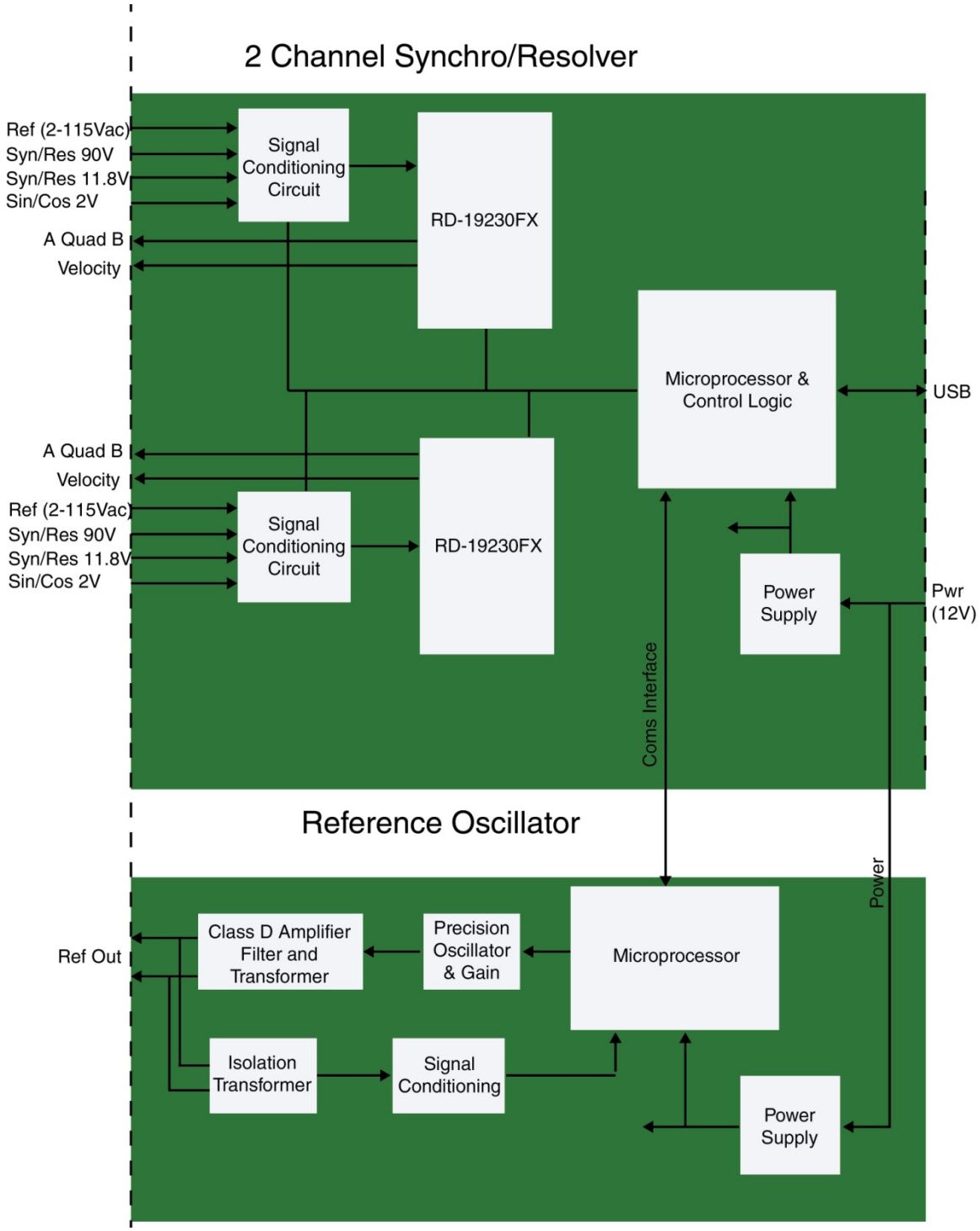


Figure 2. SB-36610UX Block Diagram

2.3 Applications

The ***SB-36610UX*** is designed for test applications involving angle position measurements from synchro and resolver sensors. Synchros and resolvers are used in applications such as motor control, industrial automation, robotics, antenna positioning, and valve control. This device is ideal for test engineers and developers to perform lab testing on these types of applications. The portable form factor is ideal for field use and the USB interface makes it suitable for use with desktop, laptop, or tablet computers.

2.4 Specifications

Table 1. SB-36610UX Specification Table

PARAMETER	MIN	TYP	MAX	UNITS
RESOLUTION (Note 2)		10, 12, 14, 16		bits
ACCURACY (Note 3)				
Carrier frequency < 5 kHz	2.3 +/- 1 lsb		1 +/- 1 lsb	arc minutes
Carrier frequency >= 5 kHz	4 +/- 1 lsb		3 +/- 1 lsb	arc minutes
SIGNAL INPUT				
Synchro (Note 3)	2	11.8, 90	90	Vrms
Z_{in} line-to-line (2 - 11.8V)		52k		Ω
Z_{in} each line-to-ground (2 - 11.8V)		35k		Ω
Common-mode Range			9.3	Vpeak
Z_{in} line-to-line (11.8 - 90V)		195k		Ω
Z_{in} each line-to-ground (11.8 - 90V)		130k		Ω
Common-mode Range			71	Vpeak
Resolver (Note 3)	2	11.8, 90	90	Vrms
Z_{in} line-to-line (2 - 11.8V)		140k		Ω
Z_{in} each line-to-ground (2 - 11.8V)		70k		Ω
Common-mode Range			9.3	V
Z_{in} line-to-line (11.8 - 90V)		520k		Ω
Z_{in} each line-to-ground (11.8 - 90V)		260k		Ω
Common-mode Range			130	V
Sin/Cos (Note 3)	0.34	2	2	Vrms
Z_{in} single ended		10k		Ω
Common-mode Range			n/a	V
REFERENCE INPUT				
Carrier Frequency				
Low bandwidth option (***SB-36611UX***)	47		10k	Hz
High bandwidth option (***SB-36612UX***)	360		10k	Hz
Type		Differential		
REFERENCE INPUT (Con't)				

Table 1. SB-36610UX Specification Table				
PARAMETER	**MIN**	**TYP**	**MAX**	**UNITS**
Voltage	2		115	Vrms
Z_{in} single ended		100k		Ω
Z_{in} differential		200k		Ω
Common-mode Range			**Note 4**	V
REFERENCE OSCILLATOR				
Option (26 Vrms)				
Voltage	2		32	Vrms
Carrier Frequency	400		10k	Hz
Resolution		1		Hz
Accuracy		0.1		Hz
Output Drive (At 26 Vrms) **(Note 5)**	1		2	VA
Option (115 Vrms)				
Voltage	20		123	Vrms
Carrier Frequency	400		10k	Hz
Resolution		1		Hz
Accuracy		0.1		Hz
Output Drive (At 115 Vrms) **(Note 5)**	1		2	VA
DIGITAL OUTPUTS DRIVE CAPABILITY				
A, B, Zero Index Pulse (ZIP)	50 pF+ Logic 0: TTL load, 1.6 mA at 0.4 V Logic 1: 10 TTL loads, = 0.4 mA at 2.8 V min. Logic 0: 100 mV max. driving CMOS Logic 1: +5 V supply minus 100mV min. driving CMOS			
POWER SUPPLY				
Voltages/Tolerances				
+12 V	10	12	15	Vdc
Current with oscillator unloaded		150		mA
Current with oscillator loaded		550		mA
THERMAL				
Device Operating Temperature	0		+71	°C
Storage Temperature	-45		+85	°C
FORM FACTOR/SPECIFICATION	USB 1.1/2.0 (Full-Speed)			
PHYSICAL CHARACTERISTICS				
Dimensions	6.38 x 5.10 x 2.17 162 x 130 x 55.1			in. (mm)
Weight	25 0.7			oz. (kg)

Table 1. SB-36610UX Specification Table				
PARAMETER	**MIN**	**TYP**	**MAX**	**UNITS**

Notes:

1. These specifications apply over the rated power supply, temperature, and reference frequency ranges;15% input signal amplitude variation, 10% reference input amplitude and 10% harmonic distortion. Refer to the RD-19230 data sheet (available at www.ddc-web.com) for specific converter specifications.
2. Resolution is software programmable for 10, 12, 14, or 16 bits.
3. Typical signal input voltage values must be used to achieve the highest accuracy. See Table 15.
4. The peak of the common mode voltage should be less than 50% of the peak of the reference input. Note that a common mode voltage may result in a degradation of accuracy and/or may cause the converter to lose track.
5. Output power of the reference oscillator derates to 1 VA maximum at the minimum voltage output.

3 HARDWARE INSTALLATION

3.1 Hardware Configuration

The ***SB-36610UX*** is a USB device, and as such does not require jumpers or switches to set the Base address or interrupt values, providing a true Plug-and-Play interface.

3.2 Connector and LED Locations

Connectors and LEDs are located on the front and rear panels of the device. Refer to Figure 3 below for specific connector and LED locations.

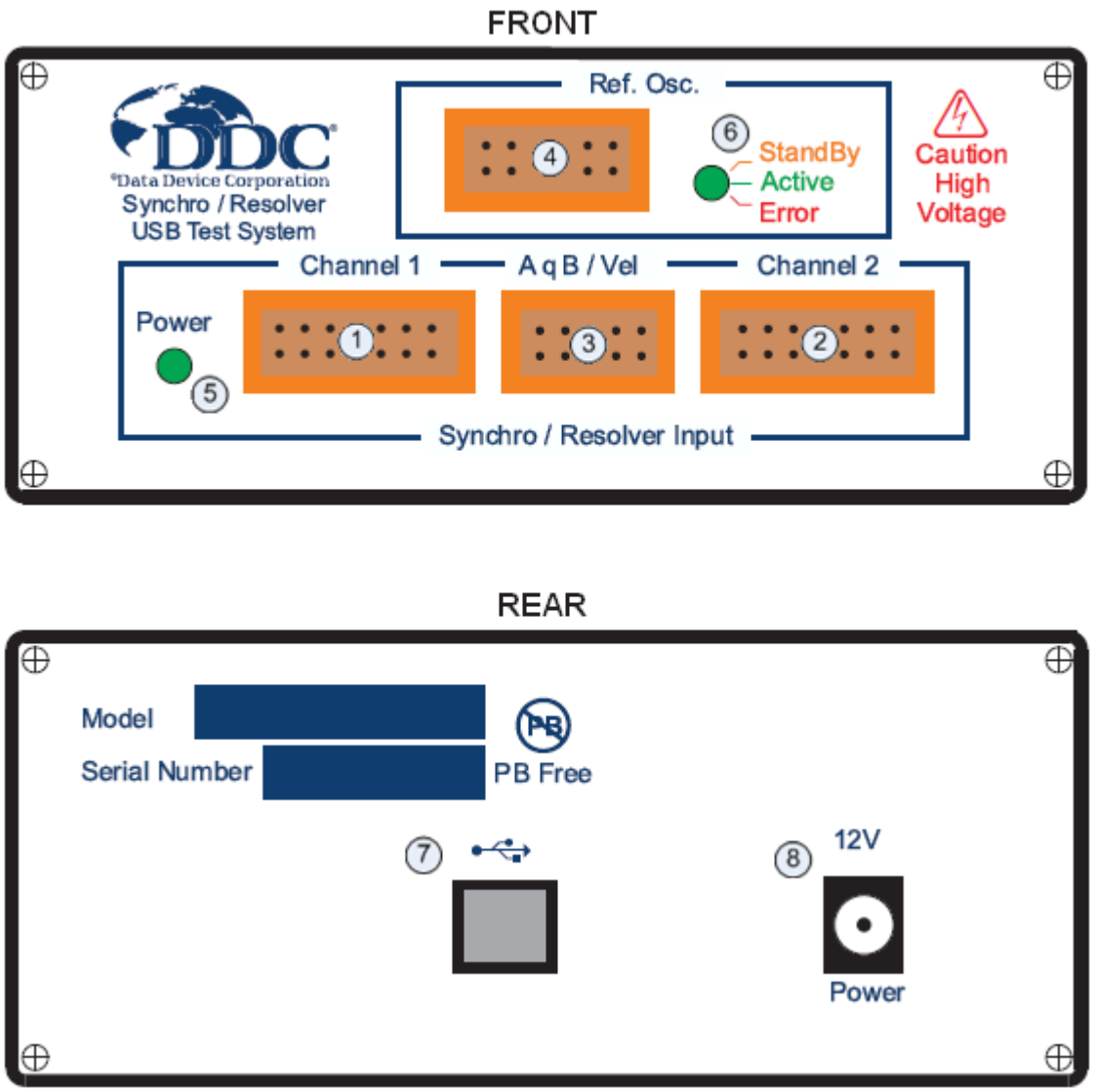


Figure 3. Connector and LED Locations

1. P1 connector for input channel 1.
2. P2 connector for input channel 2.
3. P3 connector for A Quad B and Velocity outputs for input channels 1 and 2.
4. P4 connector for on-board reference oscillator output.
5. Power indicator LED.
6. Status LED for on-board reference oscillator.
7. USB connector. Compatible with USB 1.1/2.0 interface.
8. Connector for 12V DC power input.

3.3 Module Installation/Removal

Connect the supplied AC adapter from the device to an available outlet.

Then connect the USB cable's Type B connector to the USB port of the ***SB-36610UX***, located on the rear panel of the device. After this connection is made, connect the other end (Type A) of the USB cable to a USB port in your computer.

To remove the ***SB-36610UX*** device, unplug the USB cable and AC adapter from the device.

3.4 Mounting Instructions (Optional)

The ***SB-36610UX*** device has the optional capability of mounting on a shelf or rack. The device comes with four M3 countersunk screws that are screwed into the bottom of the case. Refer to Figure 21 for the pattern dimensions of the mounting holes on the case.

To mount the device, replace each countersunk screw with a screw long enough to accommodate the thickness of the shelf or rack that it's being mounted on. See Figure 4 for calculations on the proper thread length of the screw. The rubber feet can be removed which would eliminate the 3.2 mm dimension from the equations.

Note: The removal of all four screws releases the PCB's inside of the unit. We recommend that sufficient care be taken to ensure that the PCB's do not move in this process. Also, the spacer used is made of nylon and over tightening will strip the thread.

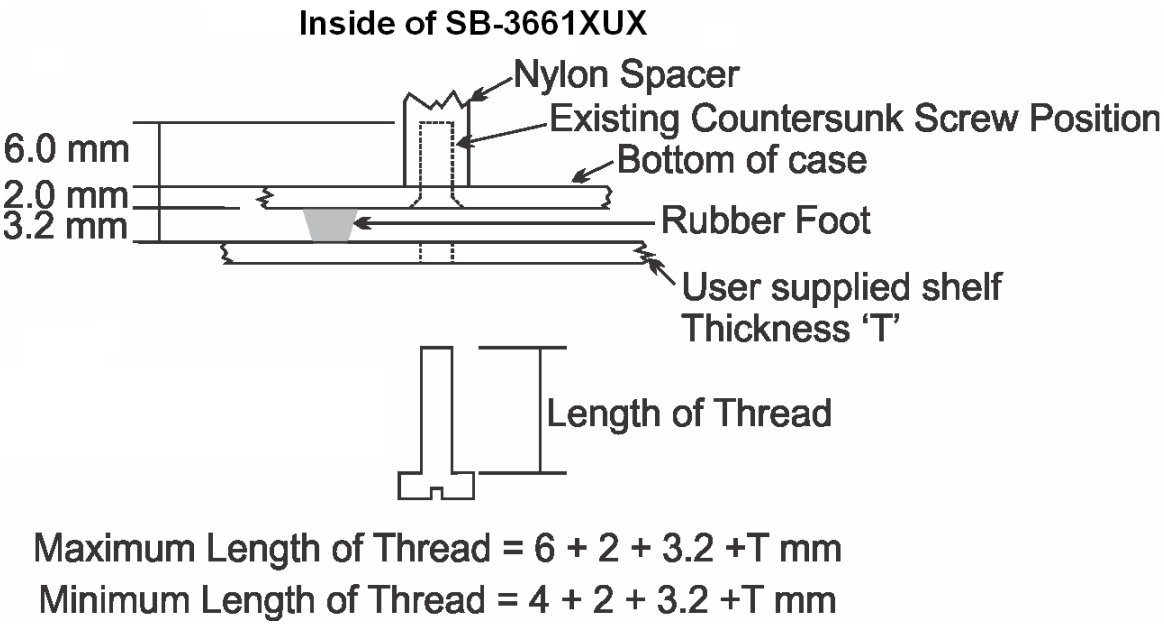


Figure 4. Mounting Diagram

3.5 Synchro / Resolver Input Signal Connections

Each input channel can accept Synchro, Differential Resolver, or Single-Ended Resolver signals. See Table 2 below for the different signal connections. Refer to Table 11 for the input channel's connector pinout.

Warning: Use caution when connecting signal inputs.

The inputs do not have overvoltage protection.

Ensure that the appropriate signal pinouts are used for the required input voltage.

Table 2. Signal Connections

Mode	Configuration			
	S1	S2	S3	S4
Synchro	X	Z	Y	No Connection
Resolver	-SIN	+COS	+SIN	-COS
Single-Ended Resolver	No Connection	COS	SIN	No Connection

3.6 Windows® XP Driver Installation

After successfully installing the hardware as described in the previous section, the driver must be installed. Before proceeding with the driver installation process, you must install the Windows software package described in section 5.2 if you have not done so already.

The following installation instructions apply to a Windows® XP operating system.

1. Turn on the computer.

2. The **"Add New Hardware Wizard"** window will appear after boot-up:

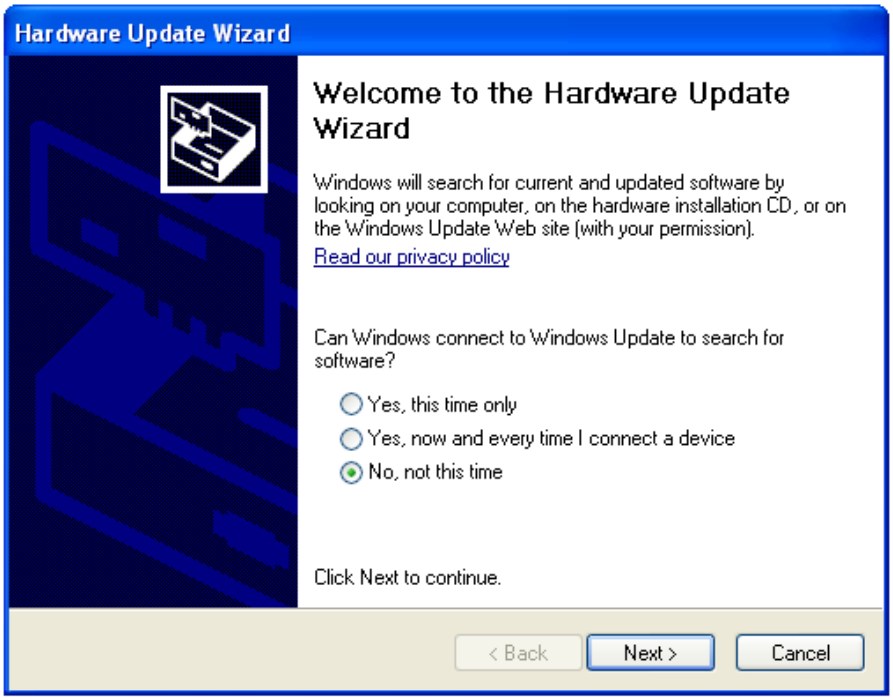


Figure 5. Windows XP - Add New Hardware Wizard Screen

3. Choose the option **"No, not this time"** and click on the **Next** button.

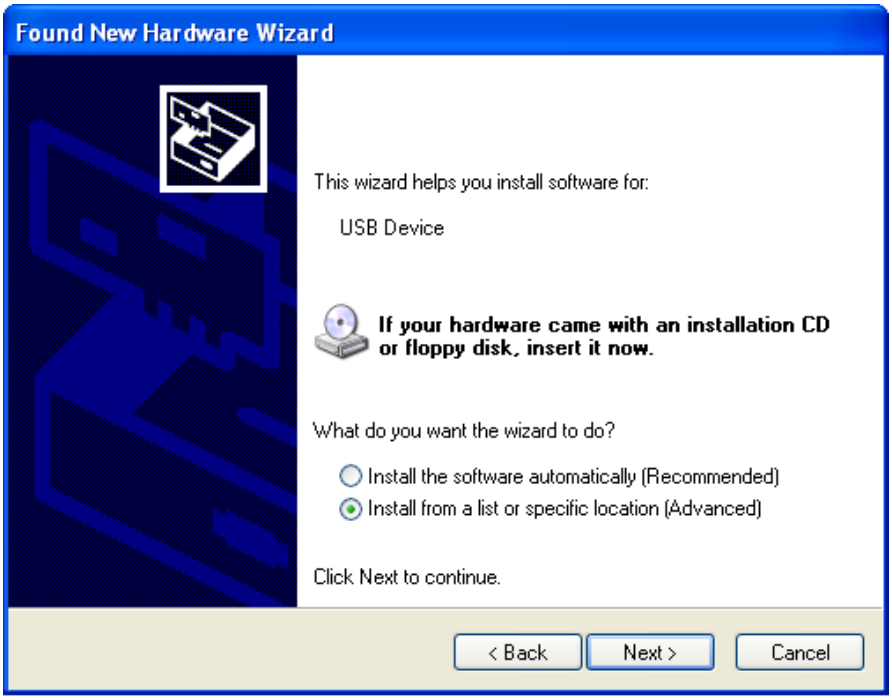


Figure 6. Windows XP - Search for New Driver Screen

4. Click on **Install from a list or specific location (Advanced)**, and click on the **Next** button.

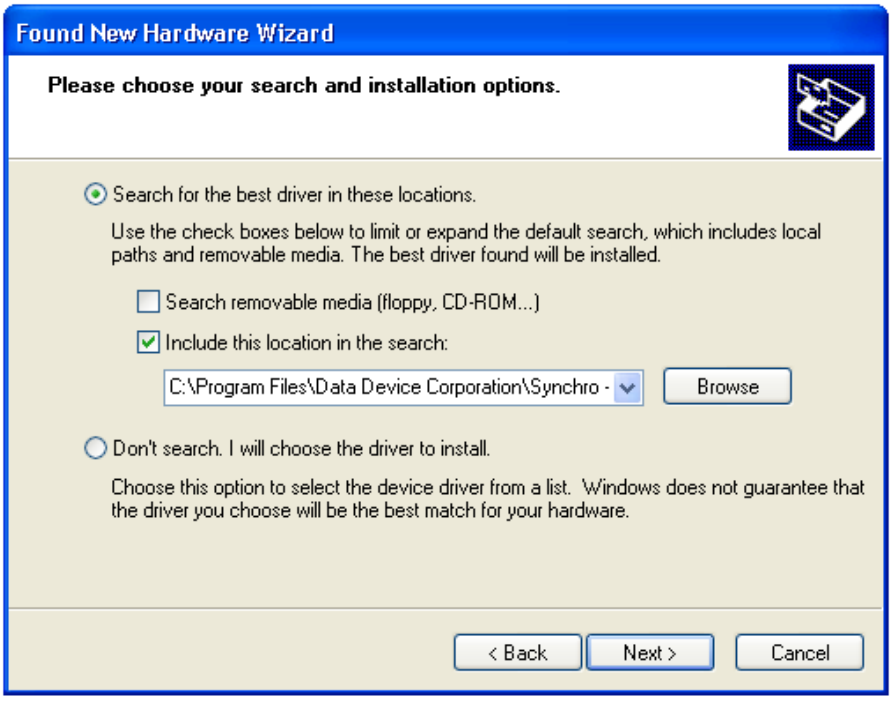


Figure 7. Windows XP - Specify Driver Location Screen

5. Select **Include this location in the search**, and click **Browse**.

6. Go to the location of where you installed the software package and choose the **Driver** directory (i.e. **C:\Program Files\Data Device Corporation\Synchro - Resolver Test System\Driver**).

7. Click on the **Next** button to locate and install the ***SB-36610UX*** device driver.

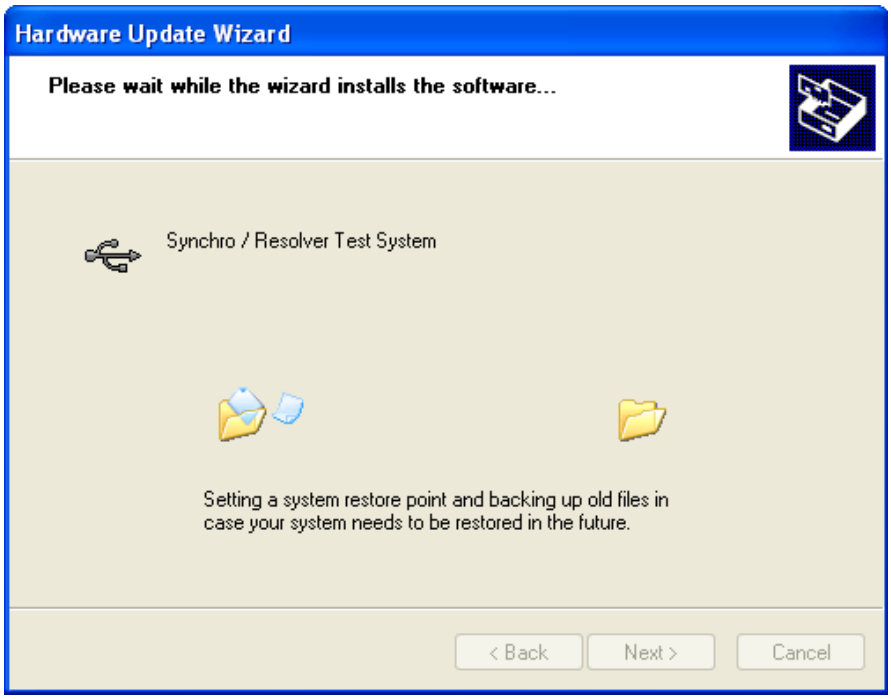


Figure 8. Windows XP - Driver Installation Screen

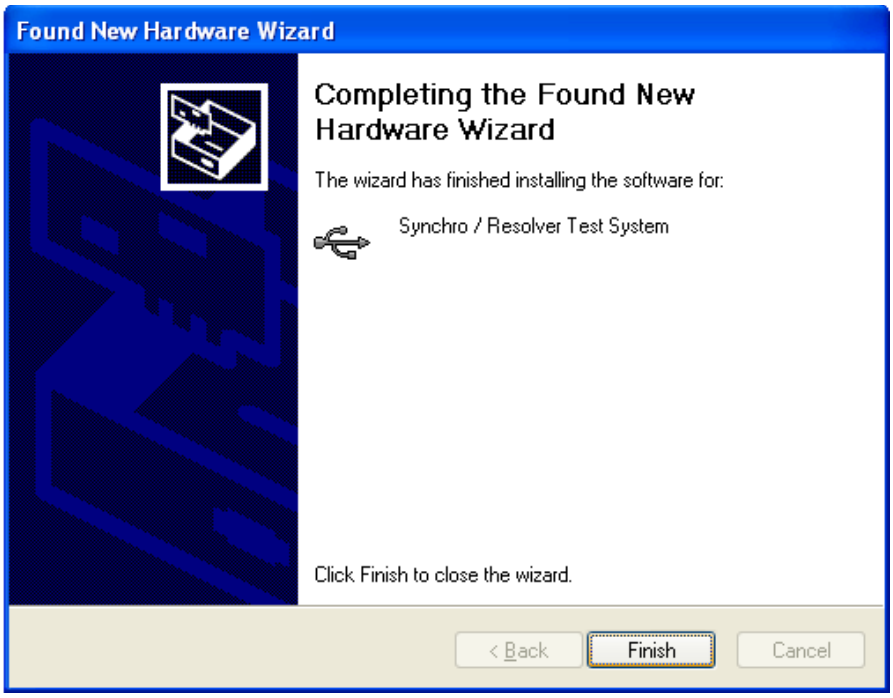


Figure 9. Windows XP - Driver Installation Complete Screen

8. Click on the **Finish** button to complete the driver installation.

3.7 Windows® Vista Driver Installation

After successfully installing the hardware as described in the previous section, the driver must be installed. Before proceeding with the driver installation process, you must install the Windows software package described in section 5.2 if you have not done so already.

The following installation instructions apply to a Windows® Vista operating system.

1. Turn on the computer.

2. The **"Found New Hardware"** window will appear after boot-up:

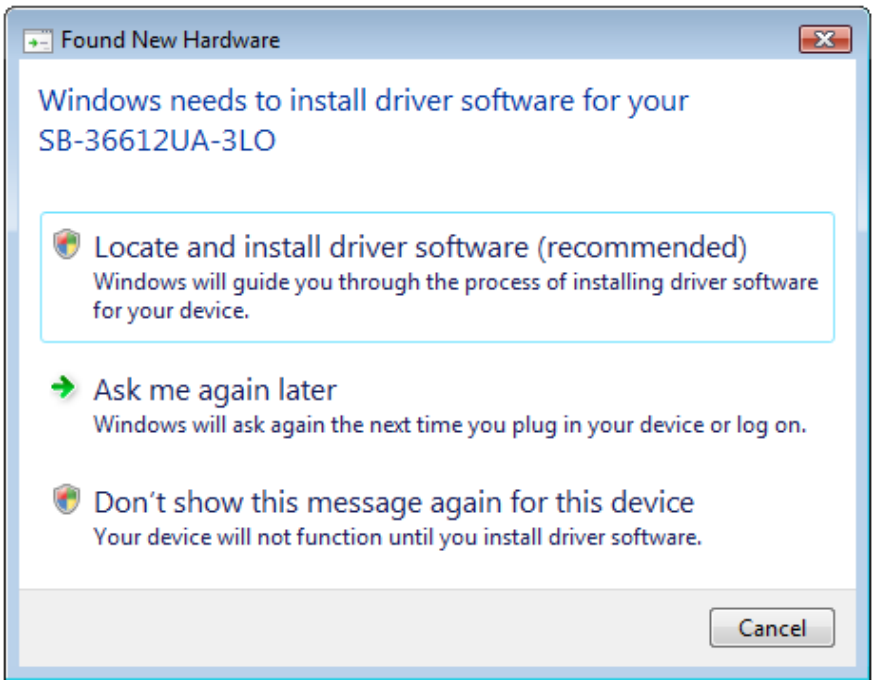


Figure 10. Windows Vista - Found New Hardware Screen

3. Click on the option **Locate and install driver software**.

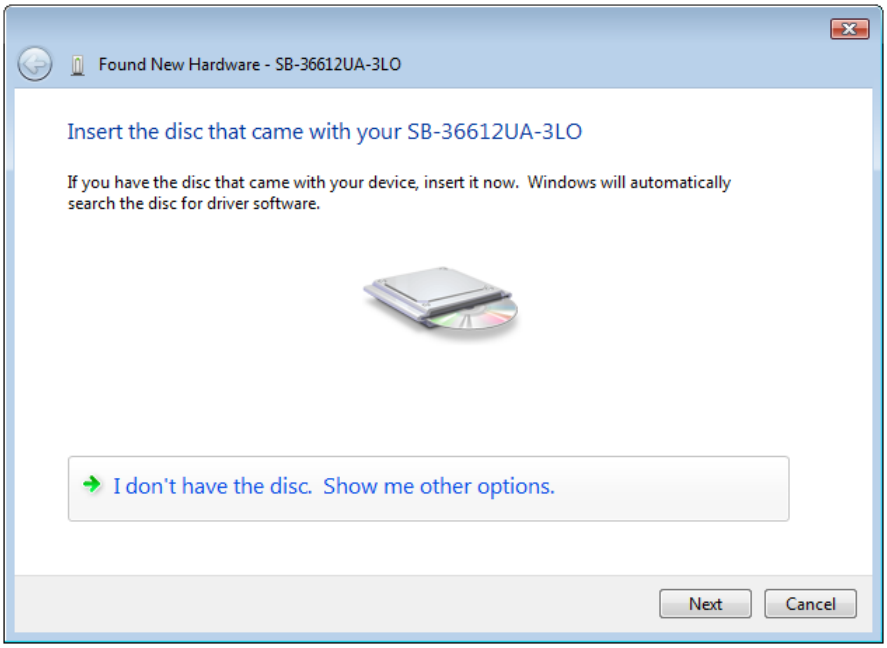


Figure 11. Windows Vista - Search for New Driver Screen (1)

4. Click on **"I don't have the disc. Show me other options."**

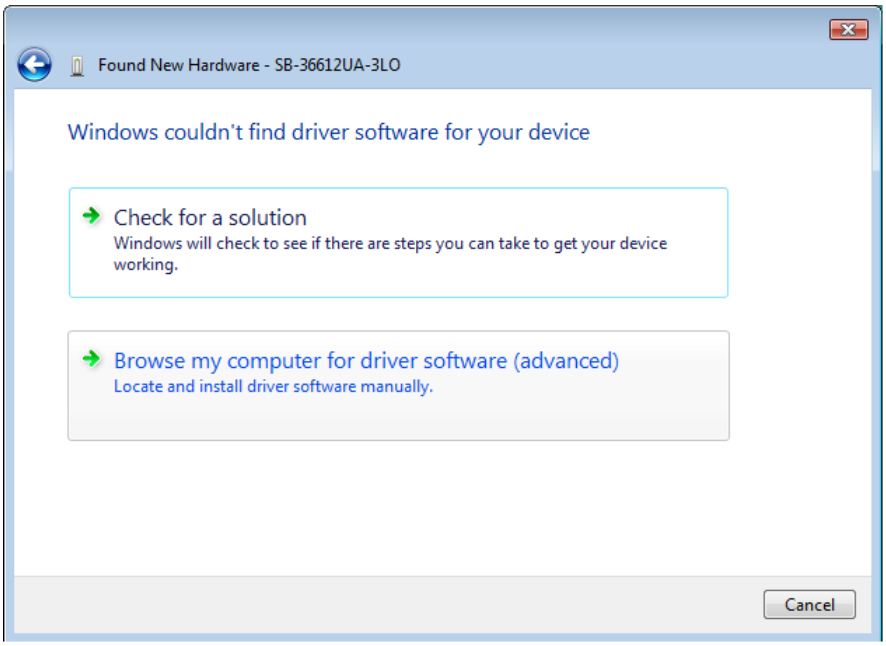


Figure 12. Windows Vista - Search for New Driver Screen (2)

5. Click **Browse my computer for driver software (advanced)**.

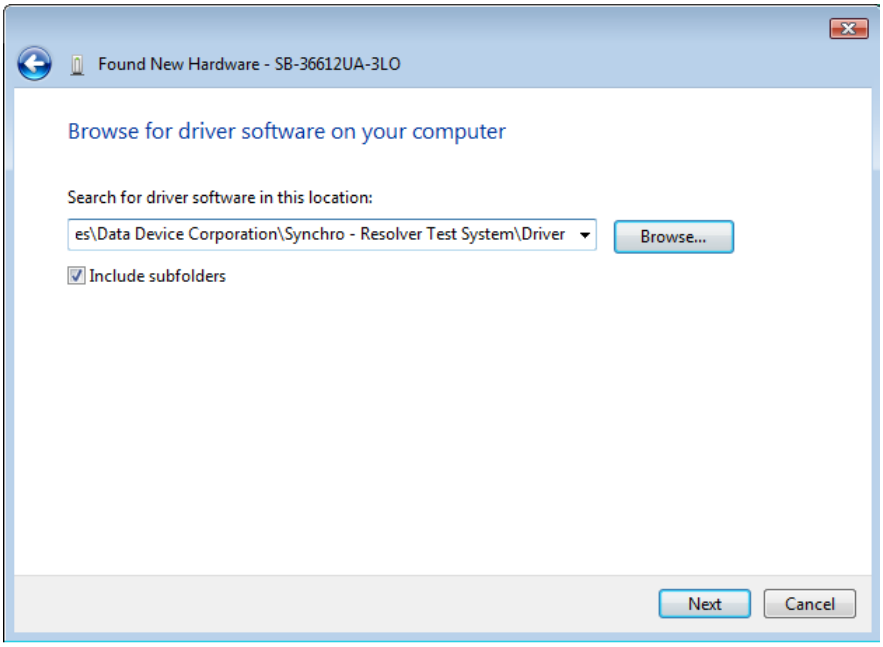


Figure 13. Windows Vista - Specify Driver Location Screen

6. Browse to the location of where you installed the software package and choose the **Driver** directory (i.e. **C:\Program Files\Data Device Corporation\Synchro - Resolver Test System\Driver**).

7. Click on the **Next** button to locate and install the ***SB-36610UX*** device driver.

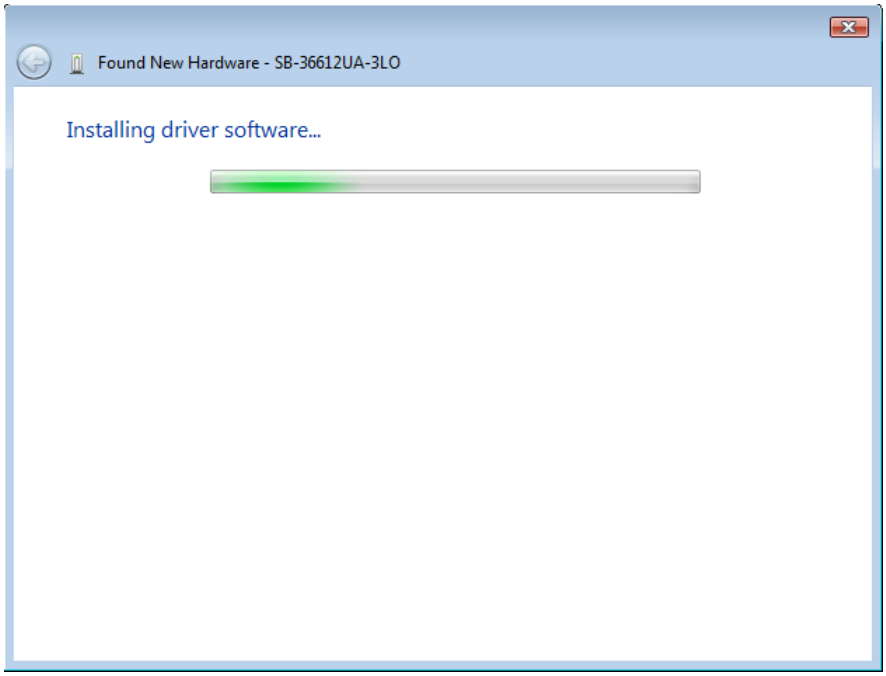


Figure 14. Windows Vista - Driver Installation Screen

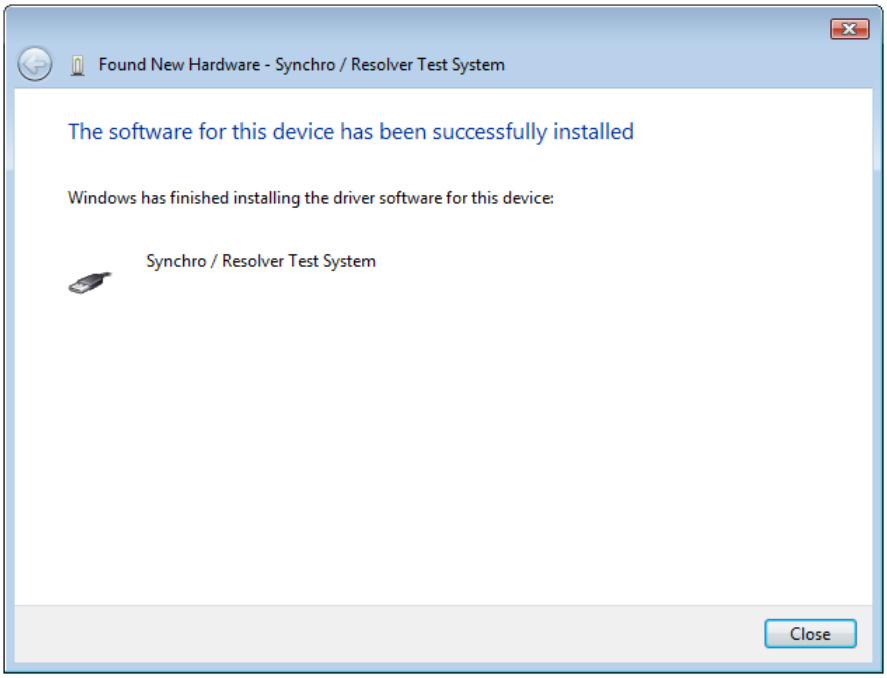


Figure 15. Windows Vista - Driver Installation Complete Screen

8. Click on the **Close** button to complete the driver installation.

3.8 Windows® 7 Driver Installation

After successfully installing the hardware as described in the previous section, the driver must be installed. Before proceeding with the driver installation process, you must install the Windows software package described in section 5.2 if you have not done so already.

The following installation instructions apply to a Windows® 7 operating system.

1. Turn on the computer.

2. You may need to manually go into the **Windows Device Manager** to install the driver.

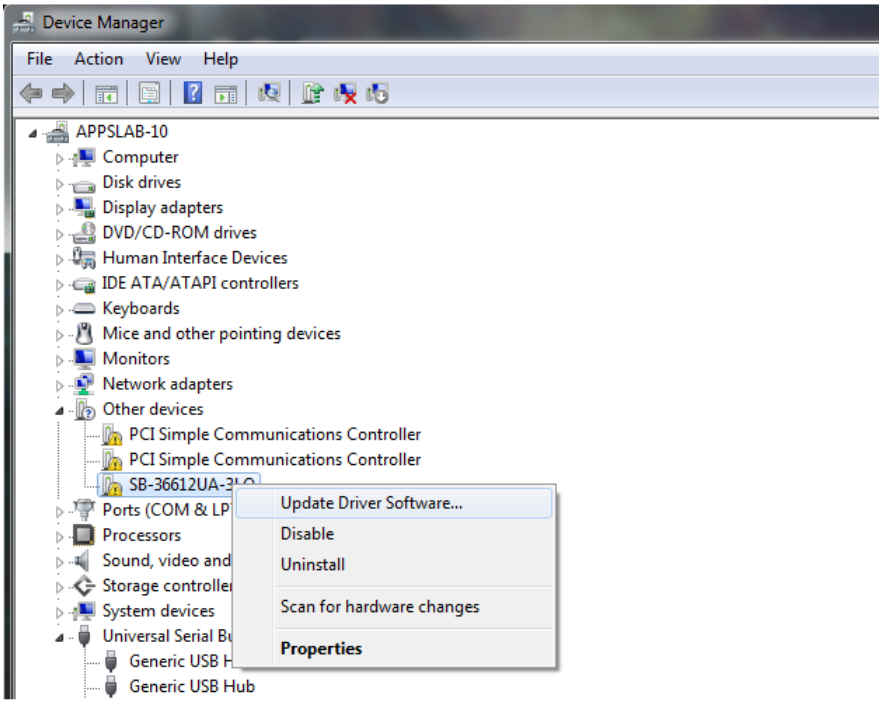


Figure 16. Windows 7 - Windows Device Manager

3. Search for the ***SB-3661XUX*** under **Other devices**, right-click on it and click on **Properties**.

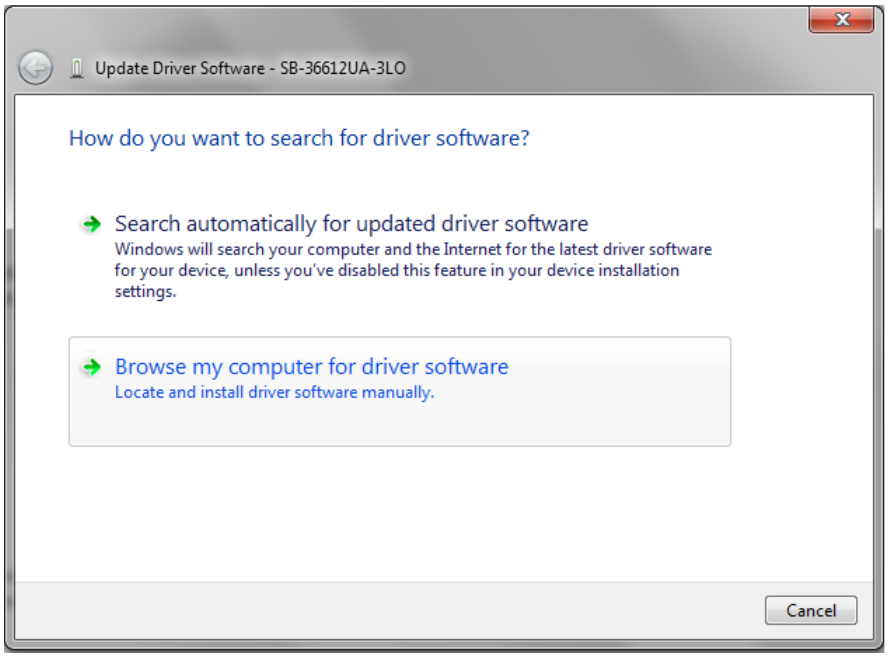


Figure 17. Windows 7 - Search for New Driver Screen

4. Click on the option **Browse my computer for driver software**.

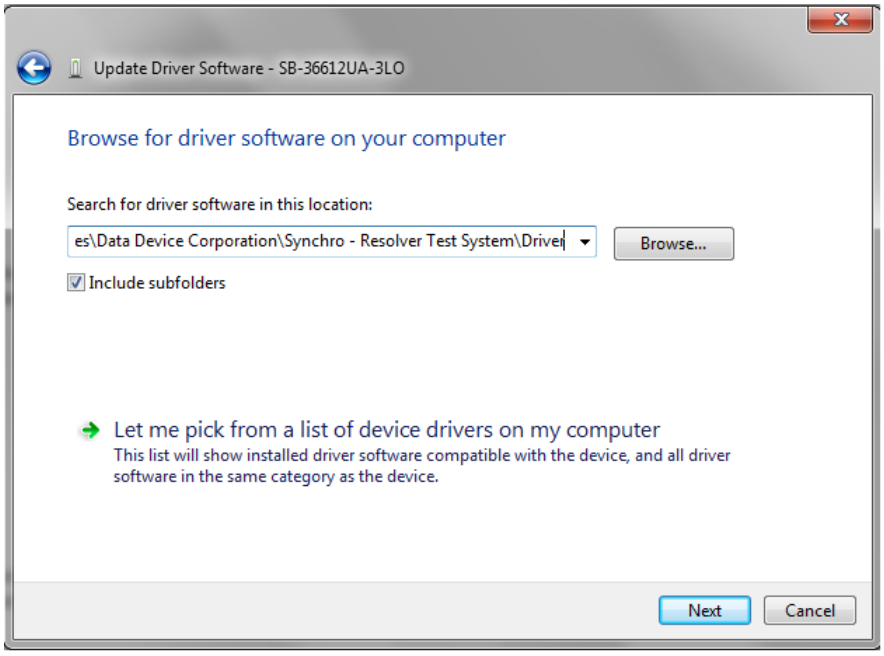


Figure 18. Windows 7 - Specify Driver Location Screen

5. Browse to the location of where you installed the software package and choose the **Driver** directory (i.e. **C:\Program Files\Data Device Corporation\Synchro - Resolver Test System\Driver**).

6. Click on the **Next** button to locate and install the ***SB-36610UX*** device driver.

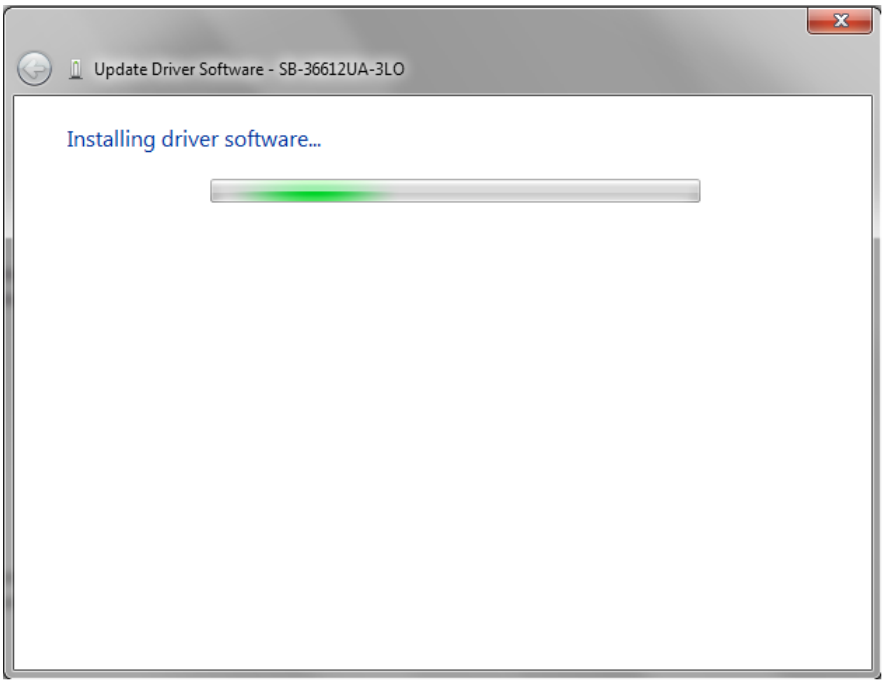


Figure 19. Windows 7 - Driver Installation Screen

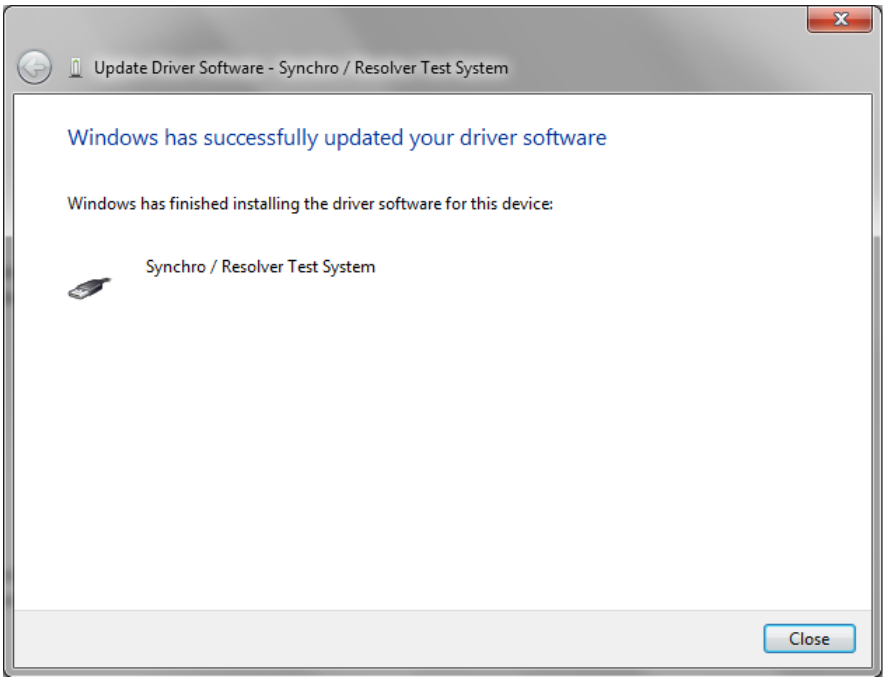


Figure 20. Windows 7 - Driver Installation Complete Screen

7. Click on the **Close** button to complete the driver installation.

4 DETAILED ARCHITECTURE

4.1 Power Supply

The ***SB-36610UX*** device is supplied with a Universal AC power adapter which will supply 12 VDC to the unit. This power adapter comes with interchangeable AC input clips to accommodate use in the following different regions:

- US
- Europe
- UK
- Australia

4.2 Mechanical Outline

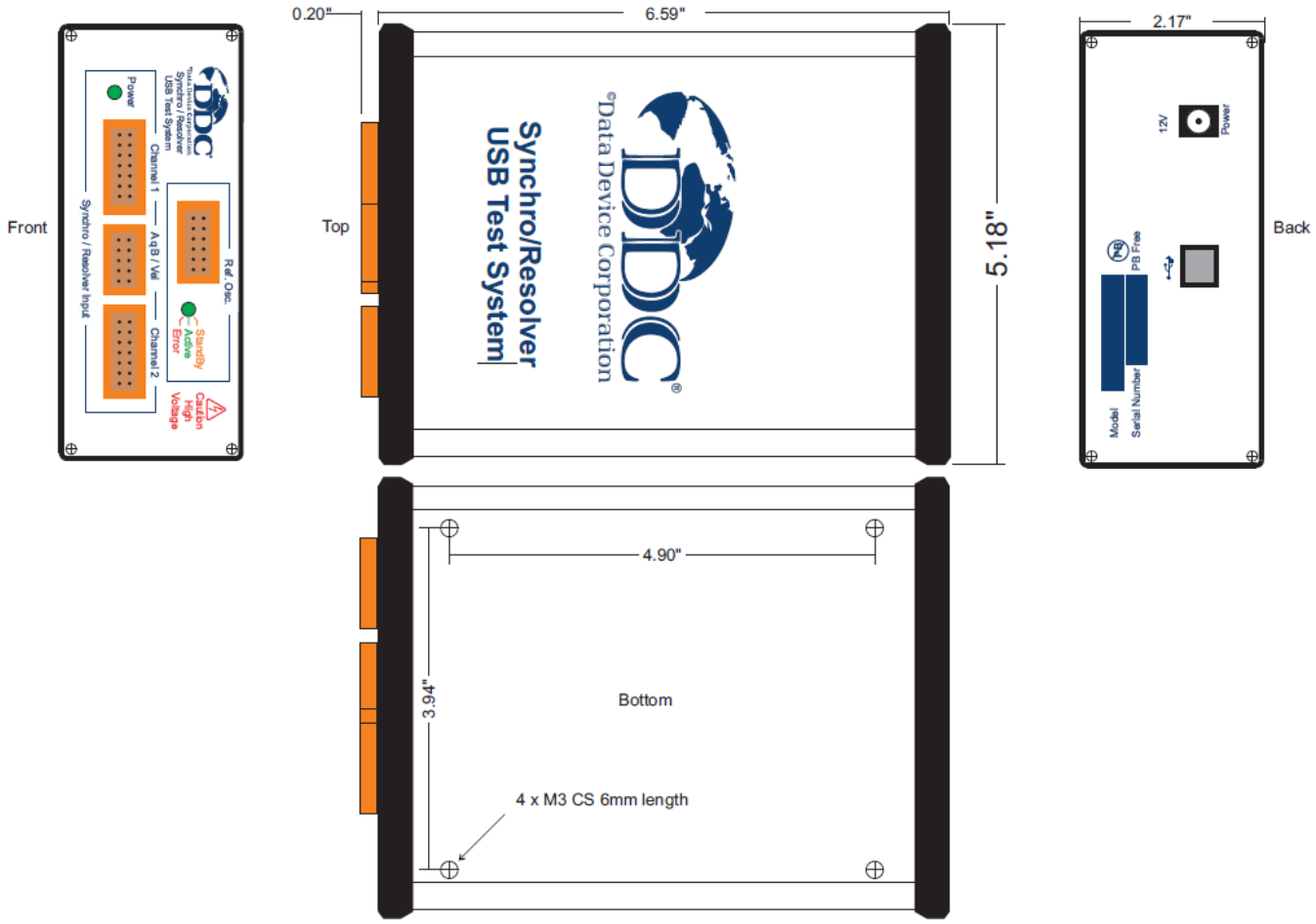


Figure 21. SB-36610UX Mechanical Outline

4.3 Synchro / Resolver to Digital Channels

4.3.1 Synchro / Resolver Interface

Each input channel can be configured through software to accept either Synchro or Resolver signals. See the **WriteSRDSignal()** function for more details.

4.3.2 Synchro / Resolver Input Signal Voltages

The ***SB-36610UX*** device has three different independent sets of signal inputs available for each channel on connectors P1 and P2. The sets are labeled based on their nominal input value. Refer to Table 11 for input channel's connector pinout.

Each set is capable of accepting a wide input signal voltage range. A user programmable software setting is provided for each set depending on what input voltage is required. These settings are configured to accept the nominal input voltages by default. Refer to the **WriteSRDGain()** function for more details.

Warning: Use caution when connecting signal inputs.

The inputs do not have overvoltage protection.

Ensure that the appropriate signal pinouts are used for the required input voltage.

Refer to Table 3 below to determine which set of inputs are to be used depending on the required interface and voltage.

Table 3. Input Voltage Ranges

P1/P2 Connector pin #'s	Interface	Nominal Input (Note 1)	Voltage Range
3-6	Synchro/Resolver	90 Vrms	15.25 – 90 Vrms
7-10	Synchro/Resolver	11.8 Vrms	2 – 11.8 Vrms
11-12	Single-ended Resolver	2 Vrms	0.34 – 2 Vrms

Table Notes:
1. Use nominal input voltage (+/- 15%) to achieve the highest accuracy.
2. See Table 15 for accuracies per voltage range.

4.3.3 Bandwidth

The user can program the device through software for each input channel independently for Low bandwidth (15/45 Hz) or High bandwidth (80/300 Hz)

depending on order selection (See Section 8). Refer to Table 4 and Table 5 for dynamic characteristics.

Use caution when operating in 10-bit and 12-bit modes. Large input steps can induce a high acceleration into the R/D converter that may cause the maximum velocity to be exceeded. If this occurs, the converter can enter a spin-around condition where it may never settle to an angle because of the low bandwidths at these resolutions.

Table 4. Dynamic Characteristics – Low Bandwidth Card (*SB-36611UX*)

Parameter	Unit	High Range (45 Hz)			
Resolution	Bits	10 (Note 1)	12 (Note 1)	14 (Note 2)	16 (Note 2)
Tracking Rate	rps. min. (typ.)	160	40	10	2.5
BW (Closed Loop)	Hz nom.	45	45	45	45
Ka	$1/sec^2$	10.13k	10.13k	10.13k	10.13k
A1	1/sec	0.25	0.25	0.25	0.25
A2	1/sec	41.03k	41.03k	41.03k	41.03k
A	1/sec	100	100	100	100
B	1/sec	50	50	50	50
Acceleration (1 LSB lag)	deg/sec	3.57k	891.4	222.9	55.7
Settling Time (179 deg. Step)	ms (typ.)	73	92	149	308
Parameter	**Unit**	**Low Range (15 Hz)**			
Resolution	Bits	10 (Note 1)	12 (Note 1)	14	16
Tracking Rate	rps. min. (typ.)	32	8	2	.5
BW (Closed Loop)	Hz nom.	15	15	15	15
Ka	$1/sec^2$	1.11k	1.11k	1.11k	1.11k
A1	1/sec	.14	.14	.14	.14
A2	1/sec	8.16k	8.16k	8.16k	8.16k
A	1/sec	33.3	33.3	33.3	33.3
B	1/sec	16.7	16.7	16.7	16.7
Acceleration (1 LSB lag)	deg/sec	390.83	97.71	24.43	6.11
Settling Time (179 deg. Step)	ms (typ.)	226	302	549	1325

Table Notes:

1. Operating at low bandwidths in low resolutions may cause the part to never settle, inducing a spin around condition. This is typically caused when inputting a large step that exceeds the maximum velocity.
2. High bandwidths in high resolutions is recommended to be used with carrier frequencies 225 Hz and above to prevent jitter.

Table 5. Dynamic Characteristics – High Bandwidth Card (*SB-36612UX*)					
Parameter	**Unit**	**High Range (300 Hz)**			
Resolution	Bits	10 (Note 1)	12 (Note 1)	14 (Note 2)	16 (Note 2)
Tracking Rate	rps. min. (typ.)	1152	288	72	18
BW (Closed Loop)	Hz nom.	300	300	300	300
Ka	$1/sec^2$	506k	506k	506k	506k
A1	1/sec	1.7	1.7	1.7	1.7
A2	1/sec	296k	296k	296k	296k
A	1/sec	711.7	711.7	711.7	711.7
B	1/sec	355.8	355.8	355.8	355.8
Acceleration (1 LSB lag)	deg/sec	178.3k	44.6k	11.1k	2.8k
Settling Time (179 deg. Step)	ms (typ.)	10.2	12.9	20.9	43.0
Parameter	**Unit**	**Low Range (80 Hz)**			
Resolution	Bits	10 (Note 1)	12 (Note 1)	14	16
Tracking Rate	rps. min. (typ.)	320	80	20	5
BW (Closed Loop)	Hz nom.	80	80	80	80
Ka	$1/sec^2$	31.6k	31.6k	31.6k	31.6k
A1	1/sec	.44	.44	.44	.44
A2	1/sec	81.6k	81.6k	81.6k	81.6k
A	1/sec	177.7	177.7	177.7	177.7
B	1/sec	88.9	88.9	88.9	88.9
Acceleration (1 LSB lag)	deg/sec	11.1k	2.8k	695	173.7
Settling Time (179 deg. Step)	ms (typ.)	40.9	51.2	81.1	161

Table Notes:
1. Operating at low bandwidths in low resolutions may cause the part to never settle, inducing a spin around condition. This is typically caused when inputting a large step that exceeds the maximum velocity.
2. High bandwidths in high resolutions is recommended to be used with carrier frequencies 1.5 kHz and above to prevent jitter.

4.3.4 Synthesized Reference

The synthesized reference eliminates errors due to phase shift within the synchro/resolver sensor of up to 45° between the reference and the signal inputs. This feature is built into both input channels of this device.

4.3.5 Incremental Encoder Emulation (A Quad B)

The device can also be used for incremental encoder emulation. The following outputs are readily available on the P3 connector: A, B, and ZIP (Zero Index Pulse). These outputs are active at all times. The timing of the A, B output is dependent on the rate of change of the synchro/resolver position (rps or degrees per second) and the encoder resolution latched into the converter (refer to Figure 22). The calculations for the timing are:

n = resolution of parallel data

$t = 1 / (2^n * \text{Velocity(RPS)})$

$T = 1 / (\text{Velocity(RPS)})$

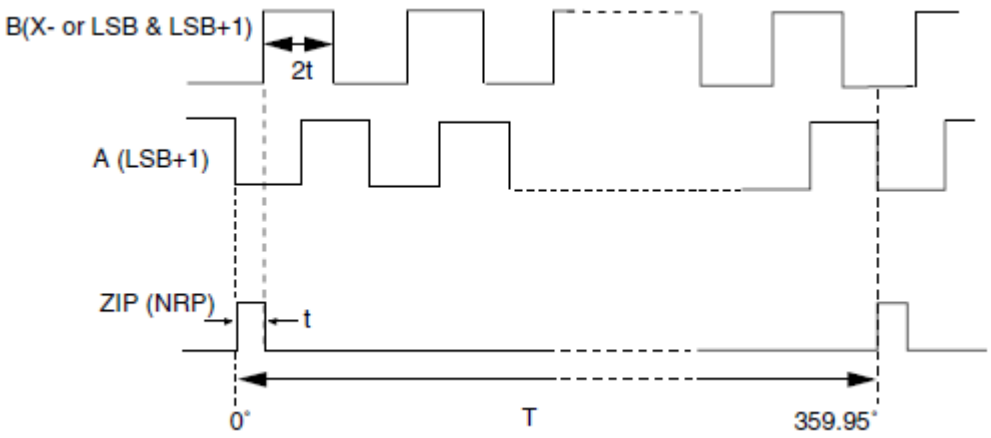


Figure 22. Incremental Encoder Emulation Timing

4.3.6 Velocity Output

Each input channel has both an analog and digital velocity output. The analog velocity output is accessible through connector P3. The voltage range is +/- 4V. The polarity indicates the direction of rotation where a positive voltage is for increasing angle. See Table 6 below for analog velocity characteristics.

Table 6. Analog Velocity Characteristics

Parameter	Units	Typical	Max
Voltage Scaling	rps/V	The rps/V scaling is dependent on resolution. See Table 4 and Table 5 for typical tracking rates.	
Scale Factor Error	%	10	20
Scale Factor TC	ppm/°C	100	200
Reversal Error	%	1	2
Linearity	%	0.5	1
Zero Offset	mV	5	15
Zero Offset TC	µV/°C	15	30
Load	kΩ	10 (min.)	

The digital velocity output is accessible through software. See Table 7 below for digital velocity characteristics.

Table 7. Digital Velocity Characteristics

Parameter	Unit	Value			
Resolution	Bits	10	12	14	16
Velocity Resolution					
***SB-36611UX* (15 Hz)**	deg/sec	0.36	0.09	0.02	0.01
***SB-36611UX* (45 Hz)**	deg/sec	1.82	0.45	0.11	0.03
***SB-36612UX* (80 Hz)**	deg/sec	4.24	0.91	0.23	0.06
***SB-36612UX* (300 Hz)**	deg/sec	12.72	3.18	0.80	0.20
Accuracy	lsb	+/- 6		+/- 4	
Tracking Rate	rps. min. (typ.)	Refer to Table 4 and Table 5 for typical tracking rates.			

4.3.7 Built-in-Test (BIT) Output

The Built-In-Test (BIT) will flag Loss-of-Signal (LOS), Loss-of-Reference (LOR), Loss-of-Tracking (LOT), and 180° phase error fault conditions. The BIT output is active low and a logical OR of these four conditions. Any one or combination of these conditions will assert the BIT output. These fault conditions are described in Table 8 below.

Table 8. BIT Fault Conditions

Fault Condition	Descritption
LOS	Both SIN and COS inputs (S1-S3, S2-S4) must fall below 0.5 Vrms.
LOR	The reference input (RH-RL) must fall below 0.5 Vrms.
LOT	This condition occurs when the difference between the analog input and digital output exceeds 100 lsbs in the positive direction or 250 lsbs in the negative direction. This typically occurs when exceeding the maximum tracking rate or during power up.
180° Phase Error	180° phase error input signal to reference input (false null) causes a BIT plus kickstarts the converter counter to correct the error.

4.3.8 Two Speed

Two speed function allows resolutions greater than 16 bits to be achieved. Refer to the Two-Speed Application Note (***AN/MFT-10***), the RD/RDC Applications Manual (***MN-19220XX-001***) and the Synchro/Resolver Conversion Handbook. These documents are available at www.ddc-web.com.

4.3.9 Self Test

The device has a built-in self test capability which can run a simulated test angle of 0, 45, or 90 degrees on each channel. Any channel not reporting back an answer within ± 1° will fail. Refer to Section 5.6, on how to setup.

4.4 On-board Reference Sine Oscillator

The on-board oscillator may be used to take the place of an external drive oscillator for the excitation signal. This oscillator is available in two options, see Table 9 for details. The oscillator frequency and voltage are programmable through software. The voltage can be programmed up to the max voltage selection ordered.

Table 9. Reference Oscillator Options

Device	Minimum	Maximum
26 Volt unit	2 Vrms	32 Vrms
115 Volt unit	20 Vrms	123 Vrms

4.5 Device Pinouts

This section delineates the user's pinouts for the ***SB-36610UX***. The connectors described here are the four connectors on the front panel.

4.5.1 I/O connector overview

The supplied mating connectors are listed in Table 10 below. For additional mating connectors, go to www.weidmuller.com.

Table 10. DDC Supplied Mating Connectors

Description	Part Number (Weidmuller)	Qty
P1/P2 Mating Connector	1727680000	2
P3 Mating Connector	1727660000	1
P4 Mating Connector	1727670000	1

4.5.2 Inserting Wires into Mating Connector

Refer to Figure 36 in the Appendix for illustrative instructions on how to insert wires into the mating connectors.

Warning: Use caution when connecting signal inputs.

The inputs do not have overvoltage protection.

Ensure that the appropriate signal pinouts are used for the required input voltage.

4.5.3 Physical Pinout Orientation for I/O Connectors

The 14-Pin P1/P2 connector receptacle shown in Figure 23 below contains the Synchro/Resolver signal inputs for Channels 1 and 2.

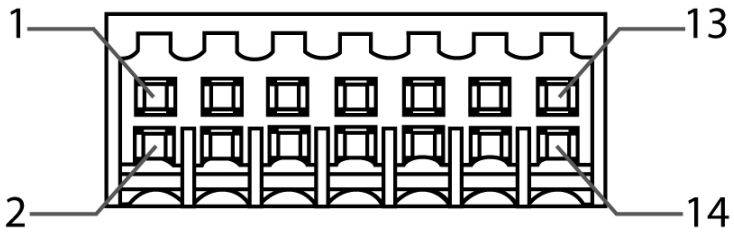


Figure 23. P1/P2 Connector

The 10-Pin P3 connector receptacle shown in Figure 24 below contains the analog Velocity and A Quad B outputs for both Channels 1 and 2.

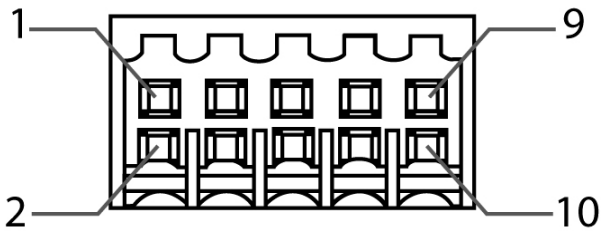


Figure 24. P3 Connector

The 12-Pin P4 connector receptacle shown in Figure 25 contains the on-board reference oscillator's outputs.

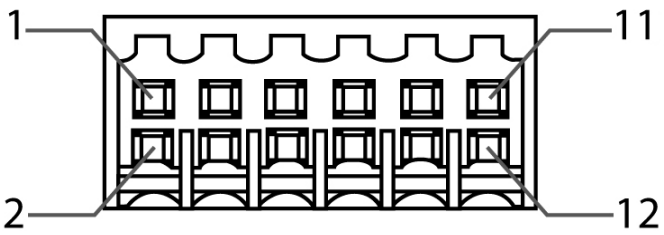


Figure 25. P4 Connector

4.5.4 Signal Naming Convention for I/O Connector

The P1 and P2 connectors are used for the inputs to channels 1 and 2 respectively. The connector's pinout is listed in Table 11.

Table 11. P1/P2 Connector Pinouts

Pin	Function	Description
1	RH	Reference excitation input high
2	RL	Reference excitation input low (respective to RH)
3	S1 – 90V	Synchro/Resolver inputs for input voltage range: 15.25 – 90 Vrms Refer to Table 2 for signal connections.
4	S2 – 90V	
5	S3 – 90V	
6	S4 – 90V	
7	S1 – 11.8V	Synchro/Resolver inputs for input voltage range: 2 – 11.8 Vrms Refer to Table 2 for signal connections.
8	S2 – 11.8V	
9	S3 – 11.8V	
10	S4 – 11.8V	
11	SIN – 2V Single Ended	Single-ended Resolver inputs for input voltage range: 0.34 – 2 Vrms Refer to Table 2 for signal connections.
12	COS – 2V Single Ended	
13	AGND	Analog ground
14	AGND	Analog ground

Table Notes:
1. All AGND pins are internally common.

The P3 connector's pinout is listed in Table 12 with a description of the signals following the table.

Table 12. P3 Connector Pinouts

Pin	Function	Description
1	VEL1	Velocity output (channel 1)
2	ZIP1	Zero index pulse output for (channel 1)
3	A1	Incremental encoder emulation output (channel 1)
4	B1	Incremental encoder emulation output; 90° Phase-shifted from "A" (channel 1)
5	GND	Ground
6	GND	Ground
7	VEL2	Velocity output (channel 2)
8	ZIP2	Zero index pulse output for (channel 2)

Table 12. P3 Connector Pinouts		
Pin	**Function**	**Description**
9	A2	Incremental encoder emulation output (channel 2)
10	B2	Incremental encoder emulation output; 90° Phase-shifted from “A” (channel 2)

Table Notes:
1. All GND pins are internally common.

The P4 connector’s pinout is listed in Table 13 with a description of the signals following the table.

Table 13. P4 Connector Pinouts		
Pin	**Function**	**Description**
1	RH	Reference excitation output high (Note 1)
2	RL	Reference excitation output low (respective to RH) (Note 2)
3	RH	(Note 1)
4	RL	(Note 2)
5	RH	(Note 1)
6	RL	(Note 2)
7	RH	(Note 1)
8	RL	(Note 2)
9	AGND	Analog ground (Note 3)
10	AGND	Analog ground (Note 3)
11	O/P En	This pin must be tied to AGND in order to enable the oscillator output.
12	AGND	Analog ground (Note 3)

Table Notes:
1. All RH pins are internally common.
2. All RL pins are internally common.
3. All AGND pins are internally common.
4. Additional pins are provided for multiple parallel device connections.

4.6 LED Indicators

The ***SB-36610UX*** has two LEDs conveniently located on the front panel to provide a user with a quick visual status of the device and the on-board reference oscillator. The power LED is located on the bottom-left and the reference oscillator status LED is

located on the top-right of the device. Refer to Figure 3 for LED locations. Table 14 below details each LED.

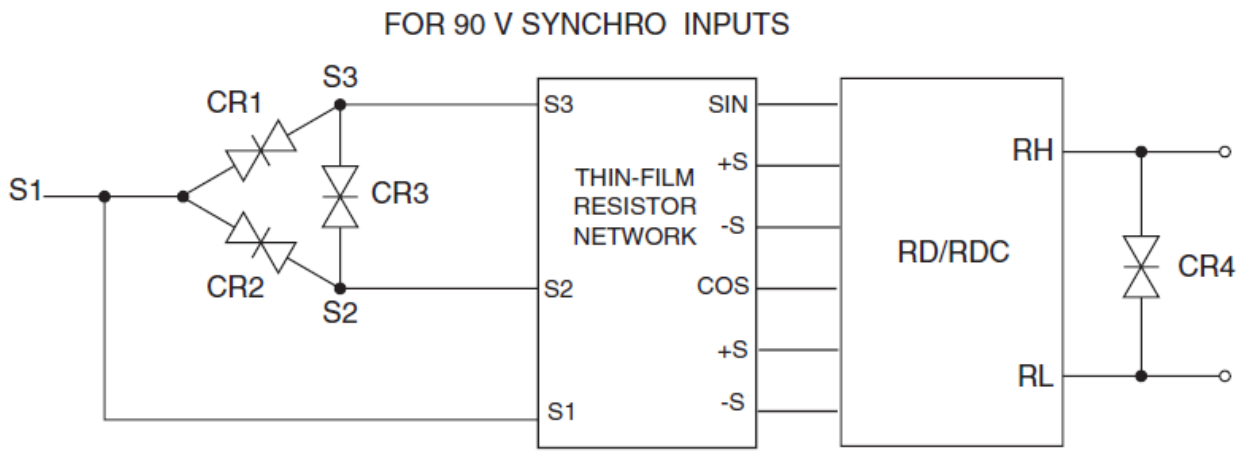
Table 14. LED Status Summary

LED	LED Color	Blinking Rate	Description
Power	Green	Steady	Normal operation.
	Red	Steady	Hardware fault exists; Contact DDC Factory (Note 1).
	Off	Steady	No power.
Reference Oscillator Status	Orange	Steady	Standby mode. Output is disabled.
	Green	2 Hz	Output is enabled.
	Red	0.5 Hz	Fault condition exists; no output.
	Off	Steady	No power.

Table Notes:
1. During the start up sequence this LED may turn red and orange for a brief moment before turning green upon applying power to the device.

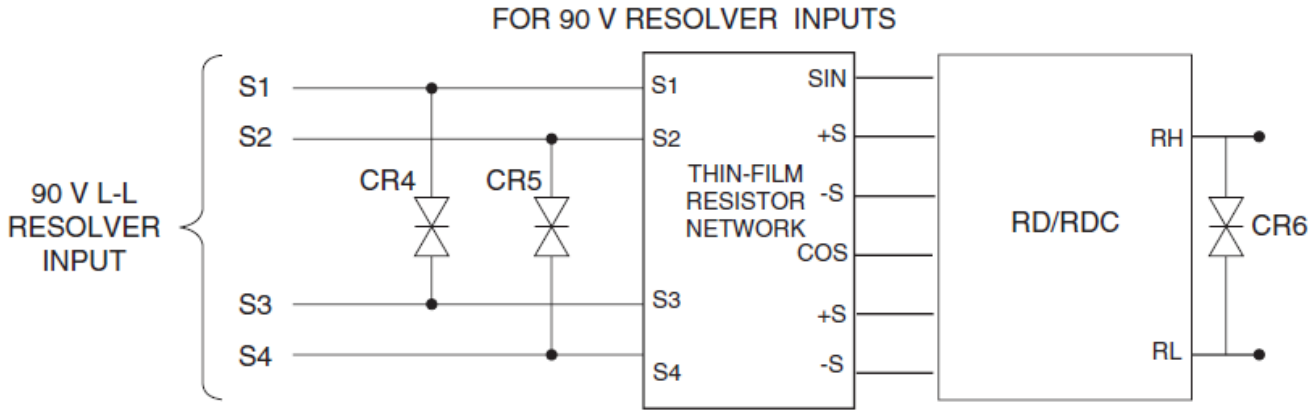
4.7 Transient Protection

Systems using the 90V line-to-line inputs may have voltage transients which exceed the maximum specification for this device's internal thin-film resistor network (500V). The 90V source may be derived from poorly regulated 115V Power Supplies, which have various high current loads. These loads may switch on and off, thus causing spikes and transients in regulation. These transients can damage the internal input thin-film resistor network. Therefore protecting these thin-film resistor networks can be done by installing voltage suppressors as shown in Figure 26 and Figure 27. Voltage transients are also likely to occur whenever the synchro or resolver input is switched on and off. For instance, a 1000V transient can be generated when the primary of a 90V control transmitter (CX) or torque transmitter (TX) driving a synchro or resolver input is opened.



CR1, CR2, CR3, and CR4 are 1.5KE170CA or 1.5KE200C-type bipolar transient voltage suppressors or equivalent.

Figure 26. Voltage Transient Suppressor, 90V Synchro Input



CR4, CR5, and CR6 are 1.5KE170CA or 1.5KE200C-type bipolar transient voltage suppressors or equivalent.

Figure 27. Voltage Transient Suppressor, 90V Resolver Input

5 SOFTWARE INSTALLATION

There are software applications which are used in conjunction with the ***SB-36610UX*** device. A software package containing the following applications are included with your device.

For Windows:

- Graphical User Interface (GUI)
- Synchro / Resolver Test System API Library (DLL)
- Command Line Interface (CLI) console application

5.1 Windows Software Overview

The ***SB-36610S0*** software package is compatible with Windows® XP/Vista/7 32-bit and 64-bit operating systems.

5.1.1 Windows GUI

An executable GUI using the SRTestSystemLib DLL is included with the ***SB-36610UX*** device to demonstrate the full capabilities of the card.

5.1.2 Synchro / Resolver Test System API Library

The Synchro / Resolver Test System API Library DLL (Dynamic Link Library) has been created to provide the user with a hardware abstraction layer for the ***SB-36610UX***. This software layer includes routines that dramatically reduce software development time by providing a high level C functions for the application programmer to interface to the USB device. Section 6 of this manual describes the routines available from the DLL.

Both 32-bit and 64-bit versions of the DLL are included with the software package:

- SRTestSystemLib_x86.dll (32-bit)
- SRTestSystemLib_x64.dll (64-bit)

5.1.3 Windows Console Application

A basic CLI console application is also included along with its source code to provide a user with example code on how to create a simple application using the API library.

5.2 Windows Software Installation

If an existing version of the software is already installed, you must first uninstall it through the **Control Panel → Add or Remove Programs**.

The Windows software is available on the MFT Software CD included with the device. To ensure you have the latest version of the software, download it off our website at www.ddc-web.com.

Perform the following steps below to install the software package for Windows from the MFT CD:

1. Insert the MFT CD into the CD-ROM drive, and allow the CD to auto start.
2. Choose your product (i.e., SB-36610UX).
3. Choose the appropriate software you wish to install (i.e., SB-36610S0). There are two separate installation files, one for 32-bit and the other for 64-bit OS.
4. Click on the Install Software icon.
5. Follow the on screen instructions to complete the installation.

Perform the following steps below to install the software package for Windows from the downloadable ZIP file:

1. Extract the .zip file.
2. Run the **Setup.exe** file for the operating system version that applies to you (x86 or x64 indicating 32-bit and 64-bit respectively).
3. Follow the on screen instructions to complete the installation.

5.3 Test System Application

The SR Test System GUI provides a user-friendly interface to the ***SB-36610UX*** device. To access the GUI, click on **Start → All Programs → Data Device Corporation → SB-3661x → SRTestSystemGUI**.

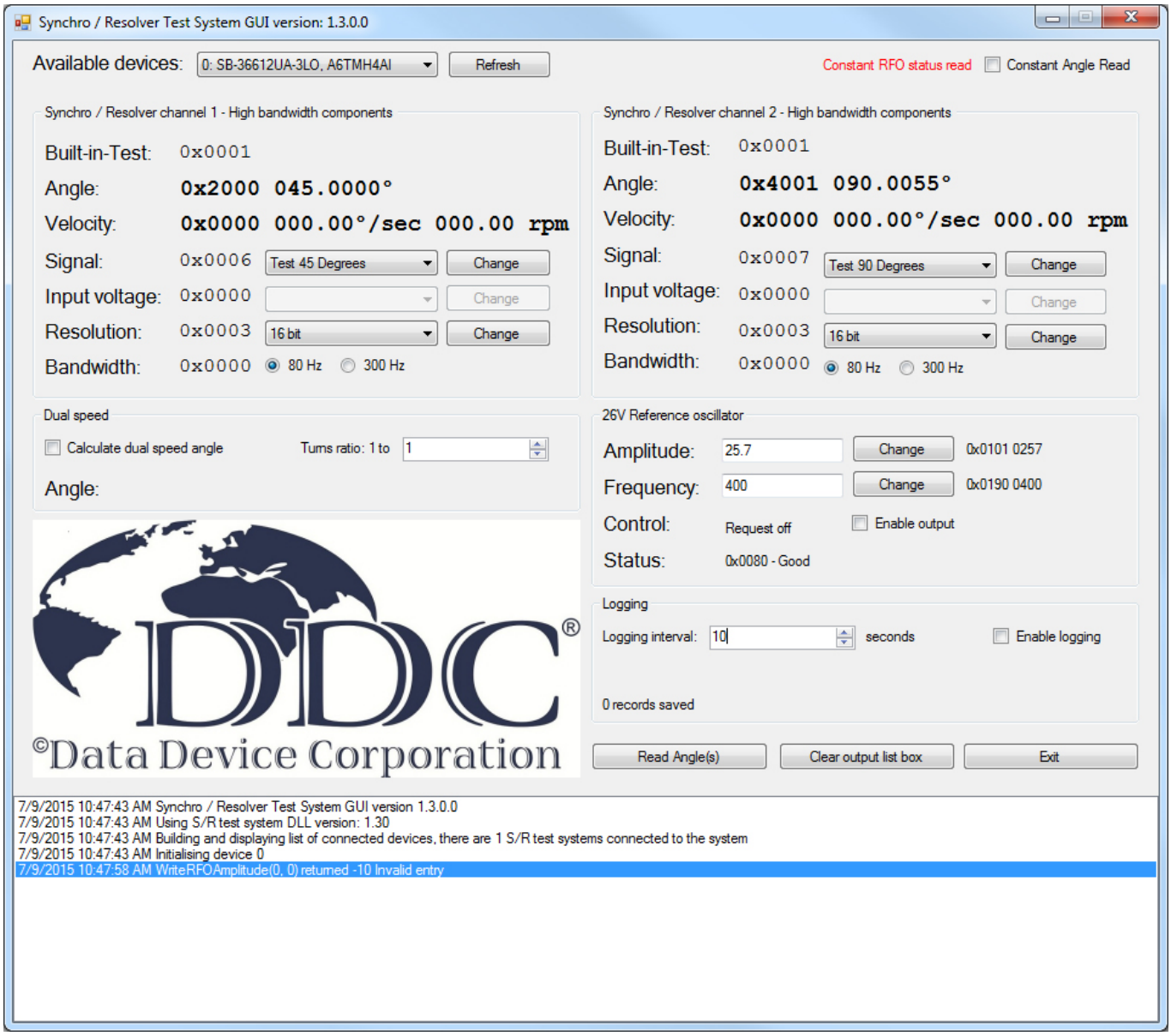


Figure 28. SR Test System GUI

5.3.1 Initialization

Upon starting the SR Test System GUI application, all installed ***SB-36610UX*** devices connected to the computer at the time will automatically be detected and initialized. You can choose which device you wish to operate with through the **Available devices** drop down list. Both the model number and serial number will be displayed for each listed device. Device # 0 is selected by default.

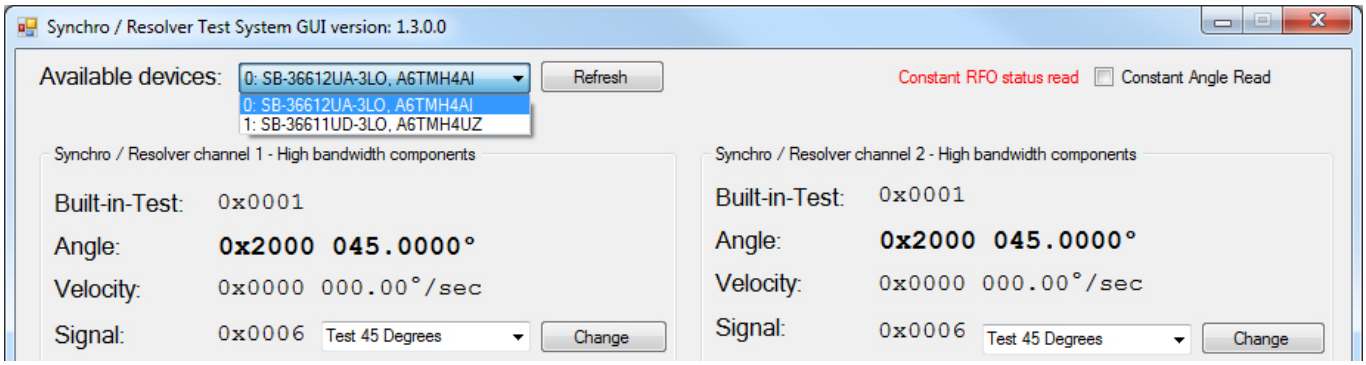


Figure 29. GUI – Selecting a Device

If additional devices are installed and connected at a later time, you must click the **Refresh** button in order to detect and initialize the device(s) before use.

Both Synchro / Resolver to Digital channels are set to the 45° self-test mode and the reference oscillator status will automatically be refreshed at a constant 300ms rate.

5.3.2 Reference Oscillator Setup

The on-board reference oscillator must be configured prior to using it. Only the amplitude and frequency need to be set. The reference oscillator label on the GUI will display either **26V** or **115V** depending on which ordering option was selected. Configuration of the on-board oscillator is not necessary when using an external excitation reference source.

To set the amplitude (Vrms), enter the desired value in decimal format (e.g. 25.7V) into the **Amplitude** field box. The max resolution is 0.1 V.

To set the frequency (Hz), enter the desired value in the form of an integer (e.g. 401 Hz) into the **Frequency** field box. The max resolution is 1 Hz.

Once a field has been modified, the **Change** button next to it will turn red. This indicates that a change was made but not yet applied. Clicking the **Change** button will apply the new value causing the button to turn back to black.

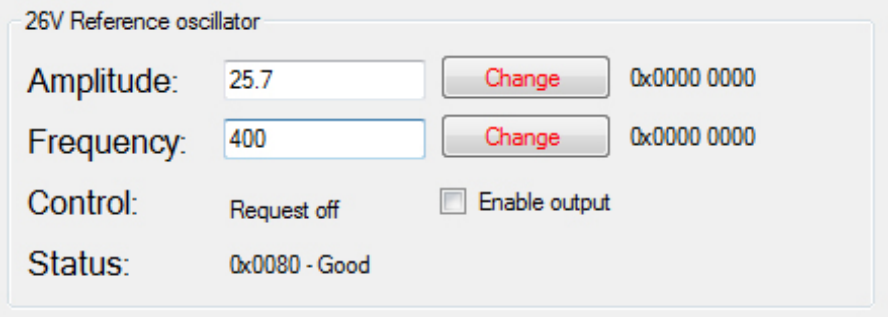


Figure 30. GUI – Reference Oscillator Configuration

A user notification will appear in the output box below if either the entered amplitude and/or frequency is outside the allowable limits.

Once the reference oscillator has been configured, the **Enable output** checkbox can be used to enable/disable the oscillator output.

A **Status** field is also provided for troubleshooting and diagnostics of the on-board oscillator. This field is automatically updated every 300ms. A detailed summary of status codes can be found on Table 16.

5.3.3 Synchro / Resolver to Digital Channels Setup

Each S/R to Digital channel must be configured prior to use. There are four configurable fields for each channel:

- Signal Mode
- Input Voltage Range
- Resolution
- Bandwidth

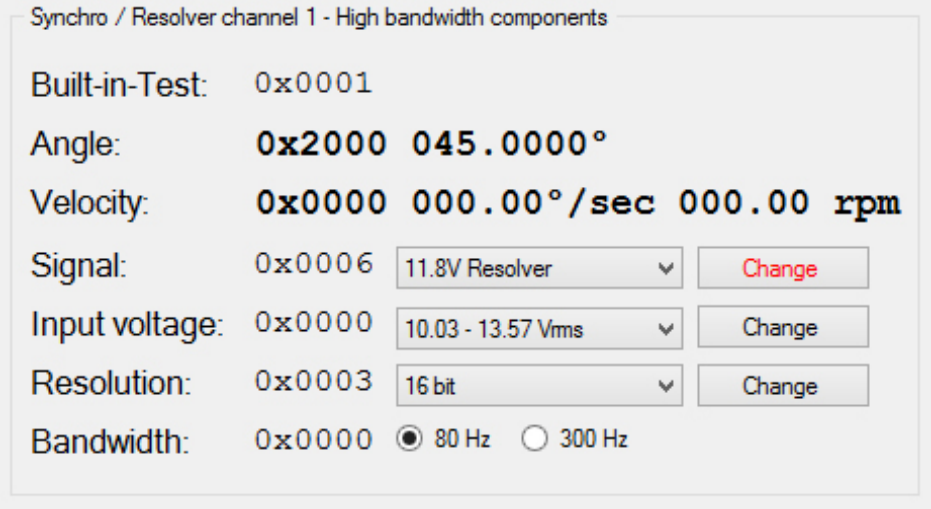


Figure 31. GUI – Channel Configuration

With the exception of the bandwidth, all configurable fields include a **Change** button that will turn red after a change in the field was made. This indicates that a change was made but not yet applied. Clicking the **Change** button will apply the new setting causing the button to turn back to black.

5.3.3.1 Signal Mode Selection

Select the desired **Signal** mode using the drop down list.

5.3.3.2 Input Voltage Selection

When you select a new **Signal** mode, the **Input voltage** will be set to the nominal range for that mode (i.e. the voltage range is set to 10.03 – 13.57 Vrms when selecting 11.8V Resolver). Use the drop down list if a different voltage range is required.

Note: *The nominal voltage range must be used in order to achieve the highest accuracy. The accuracy is de-rated when using all other voltage range selections. (See Table 15 for details)*

5.3.3.3 Resolution Selection

Select the desired **Resolution** using the drop down list. The default resolution is 16 bits.

5.3.3.4 Bandwidth Selection

Select the desired **Bandwidth** using the two radio buttons for either low or high. The bandwidth values displayed for the low and high settings are determined by the device's model number. The default bandwidth is low.

Note: *Bandwidth (BW) should be set for at least ¼ of the carrier frequency (F_c) for optimum dynamics. Lower BW will reduce noise issues. Higher BW will increase the maximum tracking rate but can cause jitter when exceeding the ¼ BW to F_c rule.*

5.4 Reading Angles

Once the channel(s) have been properly configured, there are two ways to perform angle reads:

1. Doing a single read using the Read Angle(s) button. This will provide a new angle readout for both channels even if only one channel is being used.
2. Doing constant reads using the **Constant Angle Read** checkbox. The delay time between each read is 300ms.

Additionally, the **Velocity** and **Built-in-Test** information will be updated during angle reads.

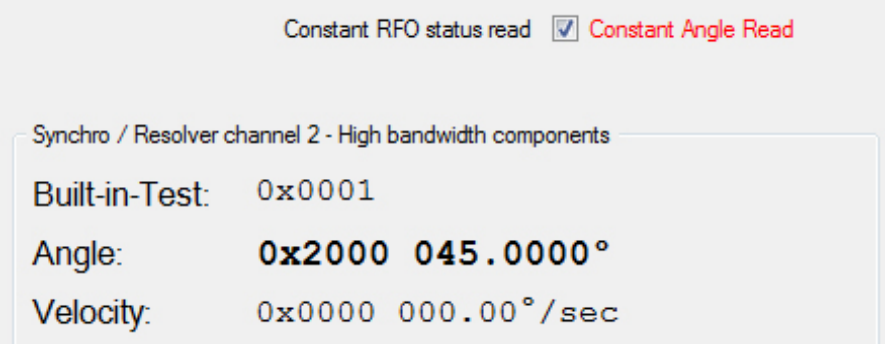


Figure 32. GUI - Angle Information

5.5 Two Speed Angles

For two speed mode, the USB box must have Channel 1 connected to the Coarse Angle and Channel 2 to the Fine Angle.

Enter the **Turns Ratio** between the Coarse and Fine Channels into the GUI.

Once the Turns Ratio has been configured, the **Calculate Dual Speed Angle** checkbox can be clicked to display the two speed angles.

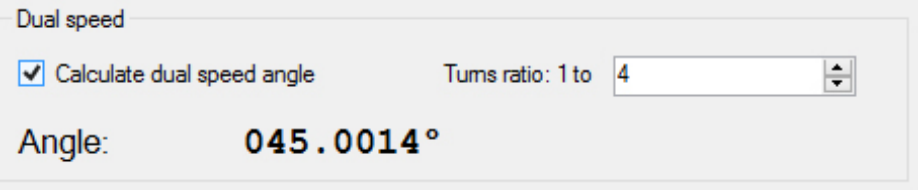


Figure 33. GUI – Two Speed Information

5.6 Self-test Feature

The device has a self-test feature for both channels and will test the device at 0, 45, or 90 degrees. If the device does not report back angles within +-1 degree, the device will fail.

Note: *The device has an internal wrap, the only connections needed to the USB device are the power supply for the USB device and USB cable to a computer.*

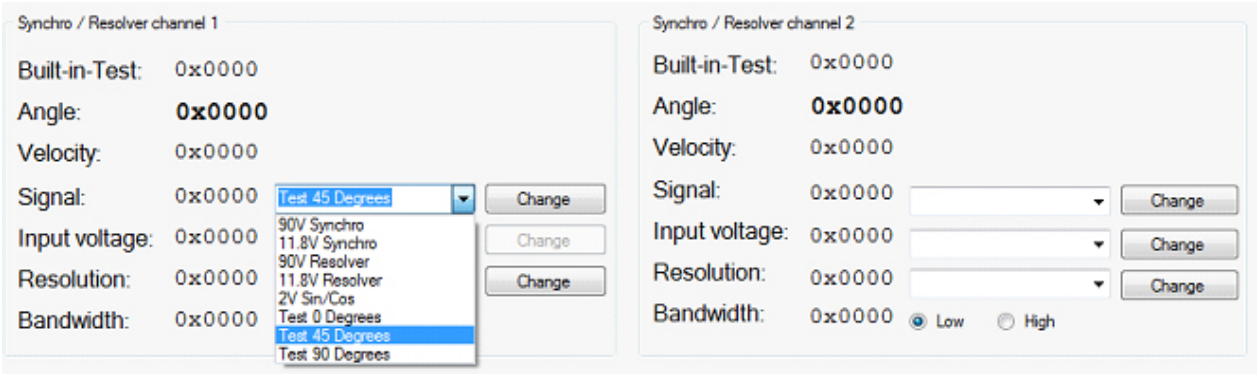


Figure 34. GUI- Self-test Feature

5.7 Data Logging Feature

The GUI has an option to log data that is being read.

1. Enable Constant Angle Read in the top right corner of GUI
2. Choose the logging interval time, this will record data every “X” amount of seconds and save the data in a desired location within a text file.

3. Check “Enable Logging” to start the data logging process

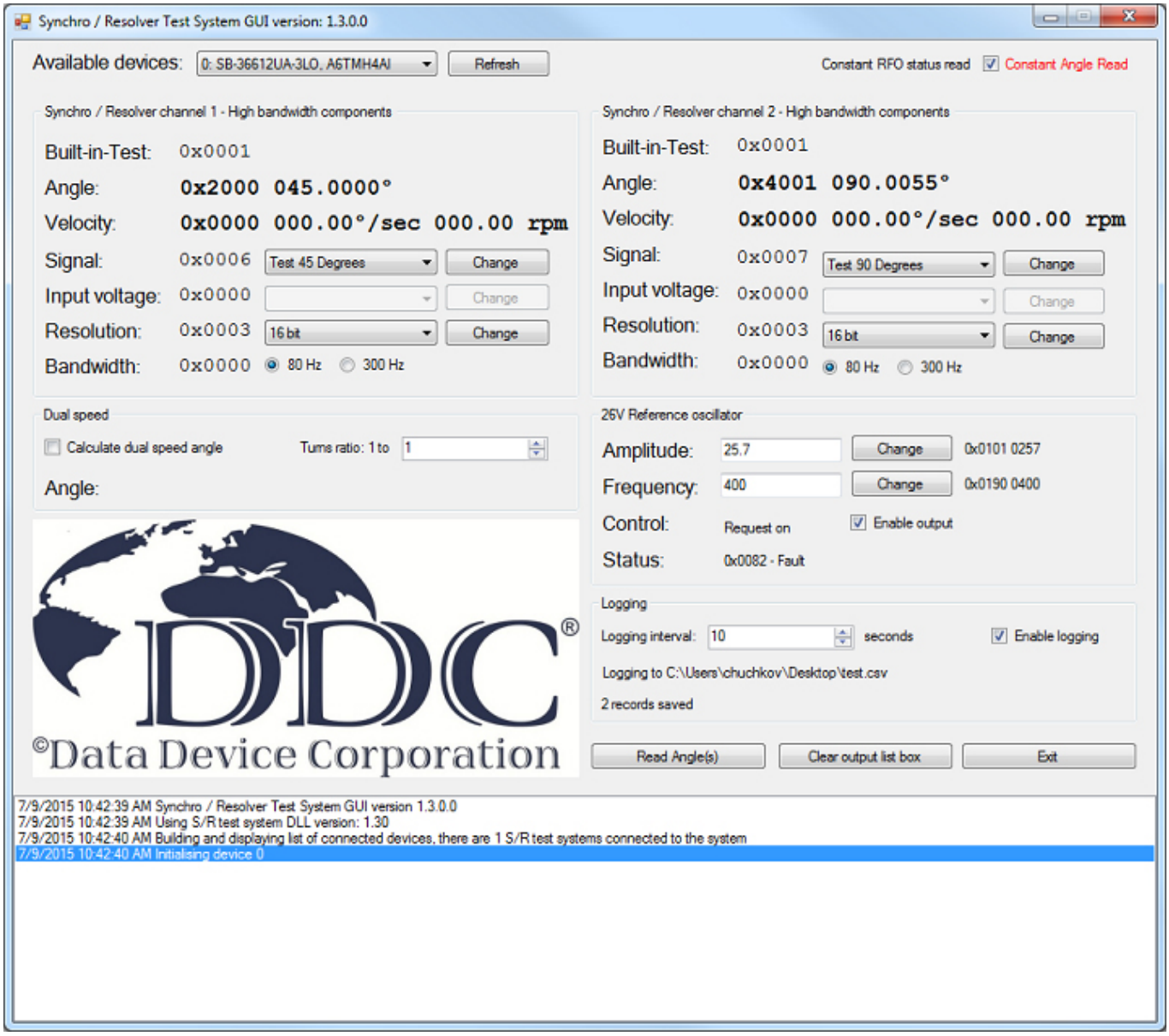


Figure 35. GUI - Data Logging Feature

6 LIBRARY FUNCTION CALLS

The following list contains all of the pertinent function calls required by the user. For any function call that returns an errorCode integer, refer to Table 17 for a detailed summary of the error codes. Any function calls contained within the SRTestSystemLib.h file that are not listed here are available from the factory upon request.

Get_DLL_Version

DESCRIPTION

Gets the version number of the Synchro / Resolver Test System DLL.

PROTOTYPE

```
int Get_DLL_Version(void) ;
```

INPUT DATA

None.

RETURNED DATA

An integer representing the DLL's version number shifted two digits to the right. (i.e. 153 indicates version 1.53).

EXAMPLE CODE

Code	Result/Returned Data
`int version = Get_DLL_Version();`	9 indicating version 0.09

BuildDeviceList

DESCRIPTION

Detect all Synchro / Resolver Test Systems connected to the computer.

PROTOTYPE

```
int BuildDeviceList(void) ;
```

INPUT DATA

None

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = BuildDeviceList();`	Detects all Synchro / Resolver Test Systems connected to the computer, and the initialises each connected device.

GetNumDevices

DESCRIPTION

Gets the number of Synchro / Resolver Test Systems connected to the computer. This function should only be called after calling BuildDeviceList.

PROTOTYPE

```
int GetNumDevices(void) ;
```

INPUT DATA

None.

RETURNED DATA

The number of Synchro /Resolver Test Systems connected to the computer.

EXAMPLE CODE

Code	Result/Returned Data
`int numDevices = GetNumDevices();`	Gets the number of Synchro / Resolver Test Systems connected to the computer.

GetSerialNumber

DESCRIPTION

Retrieves the serial number of the specified Synchro / Resolver Test System device. The serial number is an alphanumeric string of ASCII characters.

PROTOTYPE

```
char* GetSerialNumber(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

A pointer to an array of characters containing the serial number of the specified device. Maximum size 16 bytes.

EXAMPLE CODE

Code	Result/Returned Data

`char* SerialNum;`
`SerialNum = GetSerialNumber(0);`
`printf("%s", SerialNum);`
	Retrieves and prints the serial number of the first Synchro / Resolver Test System device.

GetDescription

DESCRIPTION

Retrieves the model number of the specified Synchro / Resolver Test System device.

PROTOTYPE

```
char* GetDescription(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

A pointer to an array of characters containing the model number of the specified device. Pointer to a character array. Maximum size 64 bytes.

EXAMPLE CODE

Code	Result/Returned Data

`char* ModelNumber;`
`ModelNumber = GetDescription(0);`
`printf("%s", ModelNumber);`
	Retrieves and prints the model number of the first Synchro / Resolver Test System device.

Initialise

DESCRIPTION

Sets up the communications with the specified Synchro / Resolver Test System. Should be called after connecting device to the computer and calling BuildDeviceList.

PROTOTYPE

```
int Initialise(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = Initialise(0);`	Initializes first Synchro / Resolver Test System and checks error code.

WriteSRDSignal

DESCRIPTION

Set the mode of operation of a Synchro / Resolver to Digital channel. Input checks are performed on the channel and value parameters. After a successful write, all registers are read back and checked to make sure the Synchro / Resolver Test System has acted on the new configuration.

PROTOTYPE

```
int WriteSRDSignal(int device, int channel, int value) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

value = An integer between 0 and 7 indicating the desired operation of the Synchro / Resolver to Digital channel.

0 = 90V Synchro
1 = 11.8V Synchro
2 = 90V Resolver
3 = 11.8V Resolver
4 = 2V Sin/Cos
5 = 0° Self test
6 = 45° Self test
7 = 90° Self test

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = WriteSRDSignal(0, 0, 0);`	Set channel 1 of first Synchro / Resolver Test System into 90 volt synchro mode and check error code.

WriteSRDGain

DESCRIPTION

Configures the Synchro / Resolver to Digital channel to accept a specific voltage range depending on which signal mode the channel is configured for determined by the WriteSRDSignal() function.

PROTOTYPE

```
int WriteSRDGain(int device, int channel, int value) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

value = An integer between 0 and 6 indicating the desired voltage range of the Synchro / Resolver to Digital channel. See Table 15 for the different input voltage ranges.

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = WriteSRDGain(0,0,0);`	Configures channel 1 of first Synchro / Resolver Test System to accept a voltage range of 10.03 – 13.57 Vrms when using the 11.8V Synchro or Resolver signal modes.

WriteSRDResolution

DESCRIPTION

Set the resolution of a Synchro / Resolver to Digital channel. Input checks are performed on the channel and value parameters. After a successful write, all registers are read back and checked to make sure the Synchro / Resolver Test System has acted on the new configuration.

PROTOTYPE

```
int WriteSRDResolution(int device, int channel, int value) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

value = An integer between 0 and 3 indicating the desired resolution of the Synchro / Resolver to Digital channel.

0 = 10 bit
1 = 12 bit
2 = 14 bit
3 = 16 bit

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = WriteSRDResolution(0,0,0);`	Set channel 1 of first Synchro / Resolver Test System to 10 bit resolution

WriteSRDBandwidth

DESCRIPTION

Set the bandwidth mode of a Synchro / Resolver to Digital channel. Input checks are performed on the channel and value parameters. After a successful write, all registers are read back and checked to make sure the Synchro / Resolver Test System has acted on the new configuration.

PROTOTYPE

```
int WriteSRDBandwidth(int device, int channel, bool value) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

value = A boolean value indicating high bandwidth setting (true) or low bandwidth setting (false).

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = WriteSRDBandwidth(0,0,false);`	Set channel 1 of first Synchro / Resolver Test System to low bandwidth mode

WriteRFOControl

DESCRIPTION

Turns the Reference Oscillator output on or off.

Note for safety reasons: When the reference oscillator output is enabled, it will automatically disable itself if no communication to the device over the USB bus occurs within every 1 second.

It is suggested to use the GetRFOStatus function in a constant rate loop of less than 1 second to keep the oscillator output enabled.

PROTOTYPE

```
int WriteRFOControl(int device, bool power) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

power = A boolean value setting the reference oscillator output to “ON” (true) or “OFF” (false).

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = WriteRFOConfig(0,false);`	Turn reference oscillator output off.

WriteRFOFrequency

DESCRIPTION

Sets the Reference Oscillator output frequency. Frequencies below 400 hertz and above 10,000 hertz will be rejected. The resolution is 1 Hz.

PROTOTYPE

```
int WriteRFOFrequency(int device, int frequency) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

frequency = An integer specifying the desired Reference Oscillator output frequency in hertz (Hz).

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = WriteRFOFrequency(0, 400);`	Set reference oscillator frequency in first Synchro / Resolver Test System to 400 Hz.

WriteRFOAmplitude

DESCRIPTION

Set the amplitude of the reference oscillator output. The allowed amplitudes depend on the type of device being used as shown in the following table.

PROTOTYPE

```
int WriteRFOAmplitude(int device, int amplitude) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

amplitude = An integer representing the desired amplitude in tenths of a volt. (i.e. 20 = 2.0 V and 257 = 25.7 V). Refer to Table 9 for amplitude limits.

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = WriteRFOAmplitude(0, 1150)`	Set reference oscillator amplitude in first Synchro / Resolver Test System to 115 volts (115 volt unit only).

ReadSRDRegisters

DESCRIPTION

Read all Synchro / Resolver to Digital registers (status, angle, velocity and configuration) and update the libraries' copy of these registers so they can be accessed with GetSRDStatus, GetSRDAngle, GetSRDVelocity, GetSRDRawVelocity, GetSRDConfiguration, GetSRDSignal, GetSRDGain, GetSRDResolution, GetSRDBandwidth and GetSRDCardBW. ReadSRDRegisters should be called before any of these functions.

PROTOTYPE

```
int ReadSRDRegisters(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = ReadSRDRegisters(0);`	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.

ReadRFORegisters

DESCRIPTION

Read all Reference Oscillator registers (status, control, frequency and amplitude) and update the libraries' copy of these registers so they can be accessed with GetRFOStatus, GetRFOControl, GetRFOAmplitude and GetRFOFrequency. ReadRFORegisters should be called before any of these functions.

PROTOTYPE

```
int ReadRFORegisters(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

Integer error code. See Table 17 for a detailed summary of error codes.

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = ReadRFORegisters(0);`	Reads updated Reference Oscillator registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.

GetSRDStatus

DESCRIPTION

Get the status of a Synchro / Resolver to Digital channel's Built-in test (BIT). ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int GetSRDStatus(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

An integer indicating the status of the R/D converter's Built-in test (BIT). 1 indicates no fault, 0 indicates a fault. Refer to Table 8 for the different fault conditions.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  int S1 = GetSRDStatus(0,0);`
`  int S2 = GetSRDStatus(0,1);`
`  printf("Ch1 status:0x%4x",A1);`
`  printf("Ch2 status:0x%4x",A2);`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the BIT status for both channels are printed.

GetSRDAngle

DESCRIPTION

Get the angle read by a Synchro / Resolver to Digital channel's R/D converter. ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int GetSRDAngle(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

An integer in between 0x0000 and 0xFFFF.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  int A1 = GetSRDAngle(0,0);`
`  int A2 = GetSRDAngle(0,1);`
`  printf("Ch1 angle:0x%4x",A1);`
`  printf("Ch2 angle:0x%4x",A2);`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the angle word for both channels are printed.

To convert the angle reading returned to a degree's format use the function call "Angle To." See function call description for details.

GetSRDVelocity

DESCRIPTION

Get the velocity calculated by a Synchro / Resolver to Digital channel. This value is scaled according to the channel's bandwidth and resolution settings before being returned. ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
double GetSRDVelocity(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

Any double representing the calculated velocity in degrees per second.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  int V1 = GetSRDVelocity(0,0);`
`  int V2 = GetSRDVelocity(0,1);`
`  printf("Ch1 velocity:%i",V1);`
`  printf("Ch2 velocity:%i",V2);`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the calculated velocity in deg/sec for both channels are printed.

GetSRDRawVelocity

DESCRIPTION

Get the velocity calculated by a Synchro / Resolver to Digital channel. This value is not scaled prior to being returned and it is recommended that unless there is specific reason to use this function, GetSRDVelocity should be used instead.

ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int GetSRDRawVelocity(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

A signed integer representing the value of the Synchro / Resolver to Digital channel's velocity register (-32768 to 32767).

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  int V1 = GetSRDRawAngle(0,0);`
`  int V2 = GetSRDRawAngle(0,1);`
`  printf("Ch1 velocity:0x%4x",V1);`
`  printf("Ch2 velocity:0x%4x",V2);`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the raw velocity data for both channels are printed.

GetSRDConfiguration

DESCRIPTION

Get the value of a Synchro / Resolver to Digital channel's configuration register. This register contains bits for signal, gain, resolution, bandwidth and CardBW which can all be obtained individually using their own functions (GetSRDSignal, GetSRDGain, GetSRDResolution, GetSRDBandwidth, GetSRDCardBW). Unless there is a specific reason to use this function it is recommended to use one or more of those functions.

ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int GetSRDConfiguration(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

Integer representing the value of the SRD's configuration register (including signal, gain, resolution, bandwidth and CardBW).

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  int C1 = GetSRDConfiguration(0,0);`
`  int C2 = GetSRDConfiguration(0,1);`
`  printf("Ch1 configuration:0x%4x",C1);`
`  printf("Ch2 configuration:0x%4x",C2);`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the configuration register value for both channels are printed.

GetSRDSignal

DESCRIPTION

Get a Synchro / Resolver to Digital channel's signal type (operating mode). ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int GetSRDSignal(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

Integer representing the Synchro / Resolver to Digital channel's signal type. 0 = 90V synchro, 1 = 11.8V synchro, 2 = 90V resolver, 3 = 11.8V resolver, 4 = 2V Sin/Cos, 5 = 0 degree test signal, 6 = 90 degree test signal, 7 = 45 degree test signal.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  if(GetSRDSignal(0,0) == 0)`
`  {`
`  printf("90V synchro");`
`  }`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the signal type for channel 1 is printed.

GetSRDGain

DESCRIPTION

Get a Synchro / Resolver to Digital channel's gain setting. ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int GetSRDGain (int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

Integer representing the Synchro / Resolver to Digital channel's voltage range. See Table 15 for the different input signal voltage ranges depending on the signal type.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  if(GetSRDGain(0,0) == 0)`
`  {`
`  printf("10.03 – 13.57 Vrms");`
`  }`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the voltage range setting for channel 1 is printed.
	
	Note that voltage range in the example code is based on a signal type of 11.8V Synchro/Resolver.

GetSRDResolution

DESCRIPTION

Get a Synchro / Resolver to Digital channel's resolution setting. ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int GetSRDResolution(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

Integer representing the value of the Synchro / Resolver to Digital channel's resolution bits.

0 = 10 bit
1 = 12 bit
2 = 14 bit
3 = 16 bit

EXAMPLE CODE

Code	Result/Returned Data
`int errorCode = ReadSRDRegisters(0);` `int res = getSRDResolution(0,0)`	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful. Get the resolution from channel 1.

GetSRDBandwidth

DESCRIPTION

Get a Synchro / Resolver to Digital channel's bandwidth setting. ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
bool GetSRDBandwidth(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

Boolean value, false for low bandwidth setting and true for high bandwidth setting.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  int B1 = GetSRDBandwidth(0,0);`
`  if (B1)`
`  {`
`  printf("High bandwidth setting");`
`  }`
`  else`
`  {`
`  printf("Low bandwidth setting");`
`  }`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the bandwidth setting for channel 1 is printed.

GetSRDCardBW

DESCRIPTION

Get a Synchro / Resolver to Digital channel's card bandwidth setting. This indicated the type of card (high or low) being accessed rather than the bandwidth setting which is retrieved with the GetSRDBandwidth function. ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
bool GetSRDCardBW(int device, int channel) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

channel = An integer for the channel of interest (0 for channel 1, 1 for channel 2).

RETURNED DATA

Boolean value, false for low bandwidth components (15 Hz to 45 Hz) and true for high bandwidth components (80 Hz to 300 Hz).

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  bool CB1 = GetSRDCardBW(0,0);`
`  if (CB1)`
`  {`
`  printf("High bandwidth card, 80 Hz to 300 Hz");`
`  }`
`  else`
`  {`
`  printf("Low bandwidth card, 15 Hz to 45 Hz");`
`  }`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful the device's bandwidth version is printed.

AngleTo

DESCRIPTION

Convert the angle reading returned by GetSRDAngle to a degrees, minutes and seconds format. ReadSRDRegisters() should be called before this function to read the Synchro / Resolver to Digital registers and update the libraries' copy.

PROTOTYPE

```
int AngleTo(int angle, int component) ;
```

INPUT DATA

angle = The angle to be converted into a degrees, minutes, and seconds format.

component = Integer value between 0 and 2.

0 = degrees
1 = minutes
2 = seconds

RETURNED DATA

An integer representing the degrees, minutes or seconds component.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadSRDRegisters(0);`
`if (errorCode == 0)`
`{`
`  int angle = GetSRDAngle(0, 0);`
`  int degrees = AngleTo(angle, 0);`
`  int minutes = AngleTo(angle, 1);`
`  int seconds = AngleTo(angle, 2);`
`  printf("0x%4x = %d° %d' %d\"\n", angle, degrees, minutes, seconds);`
`}`
	Reads updated Synchro / Resolver to Digital registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful, retrieve the angle from channel 1.
	
	Converts the angle to degrees, minutes and seconds and prints all values.

GetRFOStatus

DESCRIPTION

Get a Reference Oscillator's status. The meaning of each of the bits in this register is described in the table below. ReadRFORegisters() should be called before this function to read the Reference Oscillator registers and update the libraries' copy.

PROTOTYPE

```
int GetRFOStatus(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

An integer representing the Reference Oscillator status register. See Table 16 below for a bit definition.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadRFORegisters(0);`
`if (errorCode == 0)`
`{`
`  int stat = GetRFOStatus(0);`
`}`
	Reads updated Reference Oscillator registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful, retrieve the Reference Oscillator status

GetRFOControl

DESCRIPTION

Get the value of the Reference Oscillator control register. This register will be 0 when Reference Oscillator output is turned off and non-zero when it is turned on. This register can be changed with the WriteRFOControl function. ReadRFORegisters() should be called before this function to read the Reference Oscillator registers and update the libraries' copy.

PROTOTYPE

```
int GetRFOControl(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

An integer representing the Reference Oscillator control register.

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadRFORegisters(0);`
`if (errorCode == 0)`
`{`
`  int cntl = GetRFOControl(0);`
`}`
	Reads updated Reference Oscillator registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful, retrieve the current state of the Reference Oscillator control register.

GetRFOAmplitude

DESCRIPTION

Gets the currently programmed Reference Oscillator output amplitude. This value can be changed with the WriteRFOAmplitude function. ReadRFORegisters() should be called before this function to read the Reference Oscillator registers and update the libraries' copy.

PROTOTYPE

```
int GetRFOAmplitude(int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

An integer representing the current programmed amplitude in tenths of a volt. (i.e. An integer value of 257 represents 25.7 Vrms).

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadRFORegisters(0);`
`if (errorCode == 0)`
`{`
`  int amp = GetRFOAmplitude(0);`
`}`
	Reads updated Reference Oscillator registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful, retrieve the current programmed Reference Oscillator amplitude.

GetRFOFrequency

DESCRIPTION

Gets the currently programmed Reference Oscillator output frequency. This value can be changed with the WriteRFOFrequency function. ReadRFORegisters() should be called before this function to read the Reference Oscillator registers and update the libraries' copy.

PROTOTYPE

```
int GetRFOFrequency (int device) ;
```

INPUT DATA

device = An integer between 0 and 63 for the device of interest.

RETURNED DATA

An integer representing the current programmed frequency in hertz (Hz).

EXAMPLE CODE

Code	Result/Returned Data

`int errorCode = ReadRFORegisters(0);`
`if (errorCode == 0)`
`{`
`  int cntl = GetRFOControl(0);`
`}`
	Reads updated Reference Oscillator registers from first Synchro / Resolver Test System, checking the error code to determine if the read was successful.
	
	If the read is successful, retrieve the current programmed Reference Oscillator amplitude.

7 APPENDIX

7.1 Input Signal Voltage Ranges

The following is a table that defines the parameter **value** in the **WriteSRDGain()** function depending on which signal mode is selected using the **WriteSRDSignal()** function.

Table 15. Input Signal Voltage Ranges

value =	Absolute Voltage Range (Vrms)			Accuracy +/- 1 lsb (max)	
	2V (Sin/Cos)	11.8V (Synchro/Resolver)	90V (Synchro/Resolver)	Carrier Frequency < 5 kHz	Carrier Frequency >= 5 kHz
0	1.70 – 2.30	10.03 – 13.57	76.50 – 103.50	1 arc min	3 arc mins
1	1.26 – 1.71	7.46 – 10.95	56.90 – 77.00	2.3 arc mins	4 arc mins
2	0.95 – 1.27	5.55 – 7.51	42.30 – 57.30	2.3 arc mins	4 arc mins
3	0.69 – 0.94	4.13 – 5.59	31.49 – 42.61	2.3 arc mins	4 arc mins
4	0.70 – 0.52	3.07 – 4.15	23.43 – 31.69	2.3 arc mins	4 arc mins
5	0.38 – 0.52	2.28 – 3.09	17.43 – 23.58	2.3 arc mins	4 arc mins
6	0.29 – 0.39	1.70 – 2.30	12.96 – 17.54	2.3 arc mins	4 arc mins

7.2 Reference Oscillator Status

Table 16. Reference Oscillator Status Summary

Bit	Description
7:6	**Board type** “10” – 26 volt unit “11” – 115 volt unit
5	**Frequency / Voltage programming error** ‘0’ – Normal ‘1’ – Programming value error The oscillator programmed frequency and/or output voltage read by the board microcontroller from the CPLD control registers are out of allowed ranges. The flag remains set until correct values are written to CPLD control registers or the standby mode is selected.
4	**Hardware fault** ‘0’ – Hardware OK ‘1’ – A board hardware fault has been detected Logical OR of the following conditions: the microcontroller +3.3V supply voltage is out of range (+3.0V to +3.6V), microcontroller watchdog reset has occurred, SPI potentiometer setting error, microcontroller flash access error Once set, the bit will remain set if the fault condition persists
3	**Microcontroller temperature out of range** ‘0’ – Normal ‘1’ – The microcontroller temperature error The board microcontroller temperature is outside of the allowed range (-40°C and +85°C) Once set the bit will remain set if the condition persists
2	**Audio amplifier DC-DC converter fault** ‘0’ – Normal ‘1’ – DC-DC converter fault has been detected The D-Class audio amplifier DC-DC converter supply voltage is out of allowed range (+16V to 19.5V) and/or the current drawn by the converter from +5V supply rail exceeds 4A peak Once set the bit will remain set if the condition persists
1	**Oscillator output fault** ‘0’ – Normal ‘1’ – The oscillator D-Class audio amplifier has shut down possibly due to short circuit on the oscillator output Once set the bit will remain set if the condition persists
0	**Oscillator output overload** ‘0’ – Normal ‘1’ – The impedance of the load connected to the output is too low. The output voltage has been limited to a safe value Once set the bit will remain set if the condition persists

7.3 Error codes

Table 17. Error Codes Summary		
Error code	**Description**	**Details / Recommended Action**
4	I/O error	Low level driver error. Check USB connection to device and call BuildDeviceList.
3	Device not opened	
2	Device not found	
1	Invalid handle	
0	Success	Operation completed successfully. No further action required.
-1	Write timeout	Communications problem, try again.
-2	Write failed	
-3	Read timeout	
-4	Read failed	
-5	Nothing to read	
-6	Read fewer bytes than expected	
-7	Data valid still on after Synchro / Resolver to Digital configuration write	Sending of Synchro / Resolver to Digital configuration data was successful but when registers were read back it appears the Test System has not acted on the new data. Try changing the configuration again.
-8	Wrong Modbus slave address	Communications problem, data corrupted in transit. Try again
-9	Wrong Modbus CRC value	
-10	Invalid entry	One of more parameters passed to the function were invalid or out of bounds, e.g. a frequency below 400 Hz passed to WriteRFOFrequency.

7.4 Quick Reference Wiring Guide

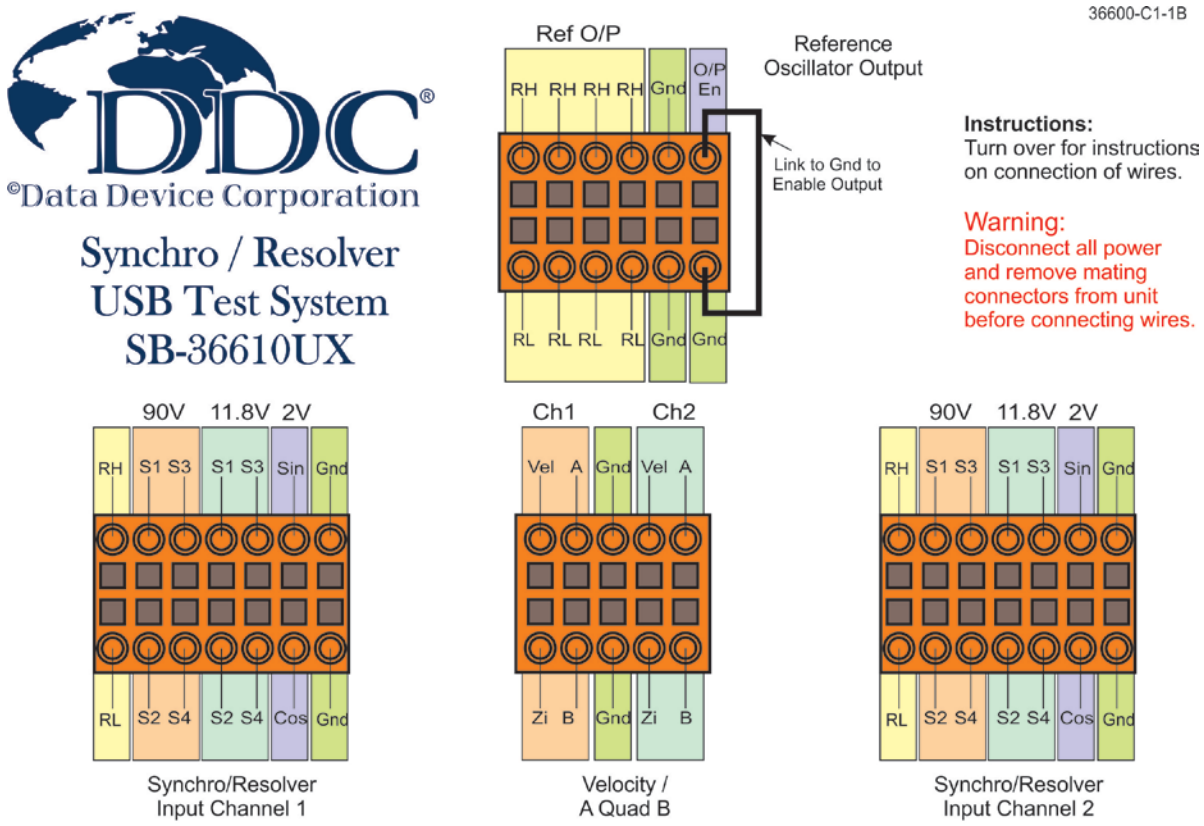


Figure 36. Quick Reference Wiring Guide (Front)

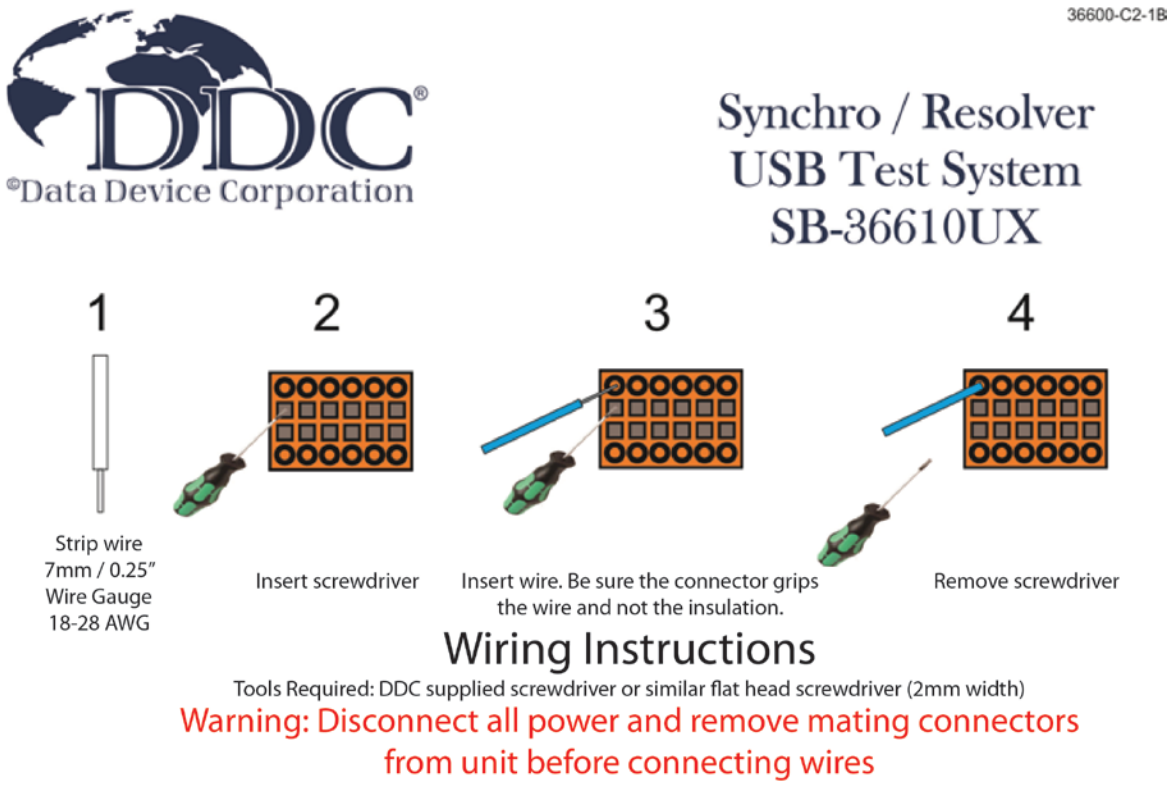


Figure 37. Quick Reference Wiring Guide (Back)

8 ORDERING INFORMATION

SB-366 1 X U X - 3L 0

Test Criteria:
0 = none

Environmental Temperature Options:
3L = 0°C to +71°C Operating Temperature Range

On-board Oscillator:
A = 26V (2 – 32 Vrms) (400 Hz to 10 kHz)
D = 115V (20 – 123 Vrms) (400 Hz to 10 kHz)

Bandwidth / Frequency Range:
1 = Low BW 15/45 Hz (47 Hz to 10 kHz)
2 = High BW 80/300 Hz (360 Hz to 10 kHz)

Device Type:
1 = Input Device (Lead-free)

Included Accessories:

- 1 USB cable
- 1 Universal AC power adapter
- 2 14-pin Mating plug-in connector
- 1 10-pin Mating plug-in connector
- 1 12-pin Mating plug-in connector
- 1 Flat head screwdriver
- 1 Quick reference wiring guide “laminated”
- Synchro/Resolver software CD
- Binary angle card

Included Software:

SB-36610SX- S/R Test System Software Package
Operating System:
0 = Windows® XP/Vista/7