Software Manual – Programming and Integration

SEA 9811

ARINC-429 Interface Module

s·e·a Science & Engineering
Applications Datentechnik
GmbH

Doc. No.:
HB/SEA 9811 Software Manual/2.1.a/Mar-2019

Address:

S.E.A. Datentechnik GmbH
Muelheimer Strasse 7
D-53148 Troisdorf

Germany

Phone: + 49 2241 12737 – 0
Fax: + 49 2241 12737 – 14

For support please contact the support address:

techsupport@sea-gmbh.com

1 Contents

2 Change Notes

#	Version	Changes
1	2.0	Update release for new LabVIEW driver architecture.
2	2.0.1	1. "Quick Start" section added. 2. ARINC Self-Test Adapter section added. 3. Module Updater chapter updated
3	2.1.a	1. Support for NI CompactRIO 904x targets added. Oldest supported LabVIEW version is now 2017. 2. Changes of formatting

3 Disclaimer, Limitation of Liability and User Responsibilities

Please carefully read the disclaimer being displayed during the driver software installation process. Do not use this product before you have read these notes.

When using or operating the S.E.A. Compact RIO Modules and the provided software, accessories, utilities or documentation in any form you must agree to the following terms and conditions.

By downloading or installing the software, by clicking on the appropriate button, you agree with the conditions in this agreement.

Please revise the latest technical information on the S.E.A. website (www.sea-gmbh.com) before installing or using the hardware and software package.

Definitions

"You" : means you as an individual, if you going to use the hardware or software provided by S.E.A. "You" also means your employer if you are going to use the software or hardware in the scope of your employment.

"S.E.A." : means the company S.E.A. Datentechnik GmbH.

"Software" : means all accompanying software including utilities and documentation delivered by S.E.A. or its authorized representatives.

"Hardware" : means all accompanying hardware components (e.g. PCB, cables, enclosures) including utilities and documentation delivered by S.E.A. or its authorized representatives.

"in Use" : Software is considered to be "in use" when loaded into any kind of memory for execution by a processor or similar device for using on a single system. This does not include a shared use via shared or networked devices.

"Compact RIO Module" and "cRIO Module" and "C Series Module" : All terms refer and mean the provided S.E.A. product for use with the RIO hardware and software platforms of National Instruments.

If you do NOT agree, you have to cancel the installation process and you can resend the hardware and software package within a period of 30 days after delivery. In this case S.E.A. will refund the product price and shipping costs.

Shipping Address is:

S.E.A. Datentechnik GmbH

Muelheimer Str. 7

D-53840 Troisdorf

Germany

General

YOU AGREE THAT YOU ARE SOLELY RESPONSIBLE FOR THE ACCURACY AND ADEQUACY OF THE SOFTWARE AND HARDWARE FOR YOUR INTENDED USE AND YOU WILL INDEMNIFY AND HOLD HARMLESS S.E.A. FROM ANY 3RD PARTY SUIT TO THE EXTENT BASED UPON THE ACCURACY AND ADEQUACY OF THE HARDWARE AND SOFTWARE IN YOUR USE. WITHOUT LIMITATION, THE HARDWARE AND SOFTWARE IS NOT INTENDED FOR USE IN THE OPERATION OF NUCLEAR FACILITIES, AIRCRAFT NAVIGATION, COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL EQUIPMENT OR MEDICAL APPLICATIONS, WHERE THE FAILURE OF THE HARDWARE OR SOFTWARE COULD LEAD TO DEATH, PERSONAL INJURY OR SEVERE PHYSICAL, ENVIRONMENTAL OR DAMAGE or PECUNIARY LOSS of ANY KIND.

Accessories

Accessories may have been supplied with the product which are specifically selected and tested for use with the product. Use of other accessories may severely degrade performance or even damage the product or other accessories, and will void the warranty. Do not try to modify, open or break the product or its components. There are no user accessible service parts inside.

Software

Follow the installation instructions as stated and described in the user hardware manual and user software manual, as well as the specific hardware operating instructions delivered with the system.

The software supplied is solely for usage along with S.E.A. provided hardware. Making copies is only valid for backup reasons. Modification or reverse engineering of the software is prohibited unless expressively provided by written agreement of an authorized S.E.A. representative or authorized by applicable laws. The software is the property of S.E.A. and is protected by international copyright treaties and statutory provisions. The user receives only the non-exclusive license to use the software for the accompanied hardware for contractual purposes. Making unauthorized copies or multiple use of the software on more than one computer system is strictly forbidden. You have especially no rights to sub-license, lease, or rent the software.

Notice

In order to help you fulfilling any warranty obligations or provide you with safety instructions, read comprehensively the hardware operating instructions (hardware manual, software manual) BEFORE operating the hardware and software.

WARNING REGARDING USE OF S.E.A. Datentechnik GmbH PRODUCTS

(1) S.E.A. PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHO'S FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS.

BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM S.E.A.s' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE S.E.A. PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY S.E.A., THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF S.E.A. PRODUCTS WHENEVER S.E.A. PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

Disclaimer and Limitation of Liability

S.E.A. Datentechnik GmbH and its subsidiaries assume no responsibility for any damage, personal injury or loss resulting from the use of this product, software and documentation.

S.E.A. Datentechnik GmbH and its subsidiaries assume no responsibility for any loss or claims by third parties which may arise through the use of the hardware, software or documentation.

S.E.A. Datentechnik GmbH and its subsidiaries assume no responsibility for any damage or loss caused by deletion of data as a result of malfunction, dead battery, or repairs. Under no circumstances shall S.E.A. Datentechnik GmbH and its subsidiaries are responsible for any loss of data or income or any special, incidental, consequential or indirect damages howsoever caused.

The contents of the documentation, manuals etc. are provided “as is”. Except as required by applicable laws, no warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose, are made in relation to the accuracy, reliability or contents of this user guide.

S.E.A. Datentechnik GmbH reserves the right to revise this disclaimer, accompanied documentation or withdraw it at any time without prior notice.

S.E.A. Datentechnik GmbH reserves the right to make changes and improvements to any hardware or software.

Only qualified and especially trained personnel are allowed to handle the hardware and software. Appropriate trainings are offered by S.E.A.

S.E.A. is not liable for damages caused by disregard of safety regulations or any other fault of a user.

This agreement terminates automatically when you fail to comply with the terms and conditions of this agreement. When this agreement terminates you must destroy all copies of the software.

All matters concerning the hardware, software or documentation delivered, contractual issues are subject to German law only. Venue is Troisdorf.

If you are unsure about the usage of the product please contact:

S.E.A. Datentechnik GmbH

Mülheimer Str. 7

53840 Troisdorf

Tel. +49 2241 127 37 0

info@sea-gmbh.com

Troisdorf, 1. March 2012

4 About This Manual

Before starting to work with the SEA 9811 please read the complete user manual and the following information carefully.

This manual is delivered as a PDF document with the possibility of full-text search. It is recommended to use the latest version of the *Adobe® Acrobat® Reader®* which you get here: http://www.adobe.com/

To make this manual more clearly specific structuring elements are used, which have the following meaning:

Names	are printed by *italic font*
[Test]	place holders are marked by squared brackets
	File path; printed by `courier font`
	Menu path
	Definition
	Example
	Prompt
	The yellow sign highlights important notes
	The blue mark highlights tips
	Reference to other documents

We believe that all information in this manual is accurate.

The document has been carefully reviewed for technical accuracy. In the event of technical or typographical errors, we reserve the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult the vendor if errors are suspected.

5 Quick Start

Please read carefully the disclaimer which is displayed during the driver software installation process. Do not use the product before you have read these notes.

This section describes how easy the SEA 9811 module can be integrated in the CompactRIO system.

Refer to the hardware manual for proper installation of the hardware.

The present (Q1/2019) the cRIO modules do not auto-discover in cRIO-904x chassis. The workaround is to manually add the modules to the LabVIEW project. NI works currently on a solution. The NI's Corrective Action Request (CAR) is #687418.

The present (Q1/2019) the cRIO modules do not auto-discover in NI-9144 and NI-9145 EtherCAT chassis. This issue only affects chassis running latest chassis firmware 16.0 and 16.1 which comes with NI-Industrial Communications for EtherCAT 16.0 and 16.1 driver. NI works currently on a solution. The NI's Corrective Action Request (CAR) #637095.

Follow the steps below to create a simple application using the SAE 9811 module and run it on a NI CompactRIO system (for this tutorial the NI cRIO-9040 has been used exemplary):

1. Ensure that you meet the hardware and software requirements listed in the previous chapter and the required software is installed on your PC and on the NI CompactRIO system.
2. Insert the SEA 9811 module into your CompactRIO system in a slot of your choice (for this tutorial slot 1 has been used).
3. Connect your CompactRIO system to the development PC via Ethernet cable and ensure a correct IP configuration of participating devices. Power up all devices.
4. Install the SEA 9811 driver software. For this VI Packet Manager (VIPM) is required. If VIPM is not yet installed on your system please start LabVIEW and select *Tools -> Find LabVIEW Add-ons...*. Complete the VIPM installation following the instructions.
5. Detect/add the NI CompactRIO system in NI-MAX Network environment (Netzwerkumgebung). Ensure that the SEA 9811 modules is configured to 'LabVIEW FPGA', refer to the screen shot below. Close NI-MAX afterwards.

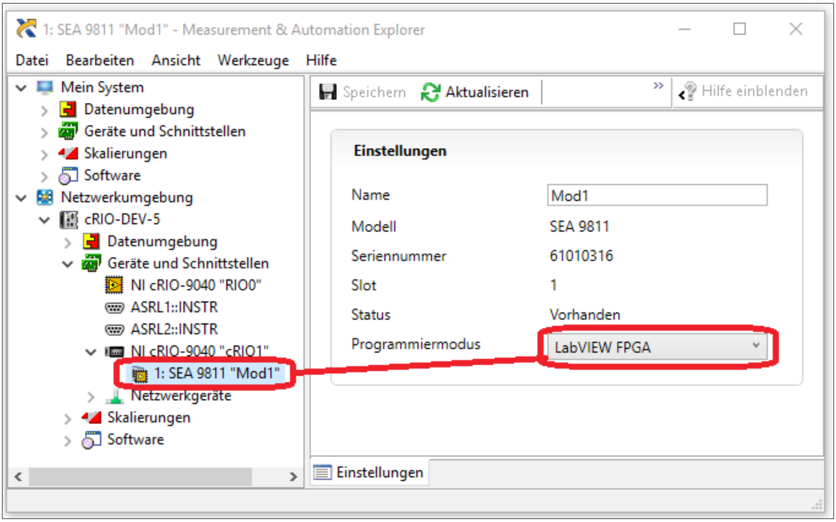


Fig. 1: Quick Start – NI-MAX configuration

6. Start LabVIEW and create a new, blank project.
7. The project explorer window appears. In this window select the uppermost item in the tree (*Project: Untitled Project 1.lvproj*) and select *New -> Targets and Devices* right-clicking on it:

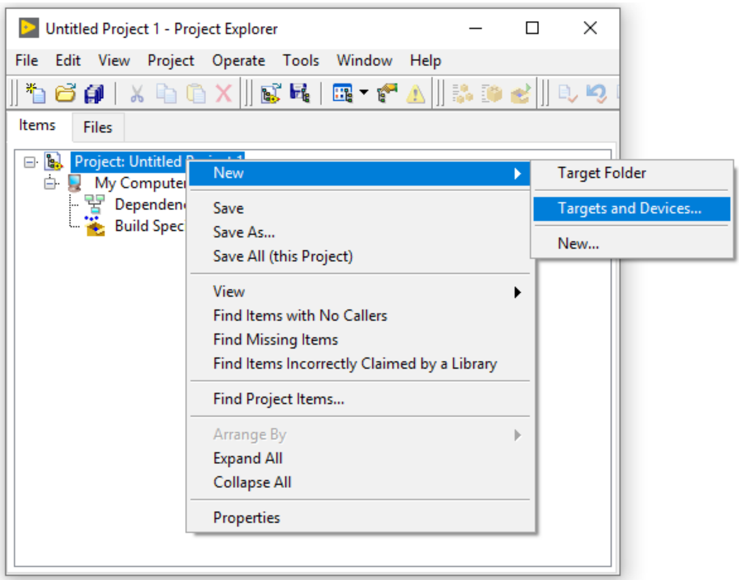


Fig. 2: Quick Start – Add a new target

8. Select your NI CompactRIO system from the list of available targets or add it manually. The chassis, FPGA target and cRIO modules are detected automatically. If automatic detection fails please add the system components (chassis, FPGA target, cRIO modules) manually to the LabVIEW project.

9. Ensure that the chassis is configured to 'LabVIEW FPGA Interface' RIO programming mode.

10. The configuration is completed. After a successful discovering the project explorer window should look similar to figure 3.

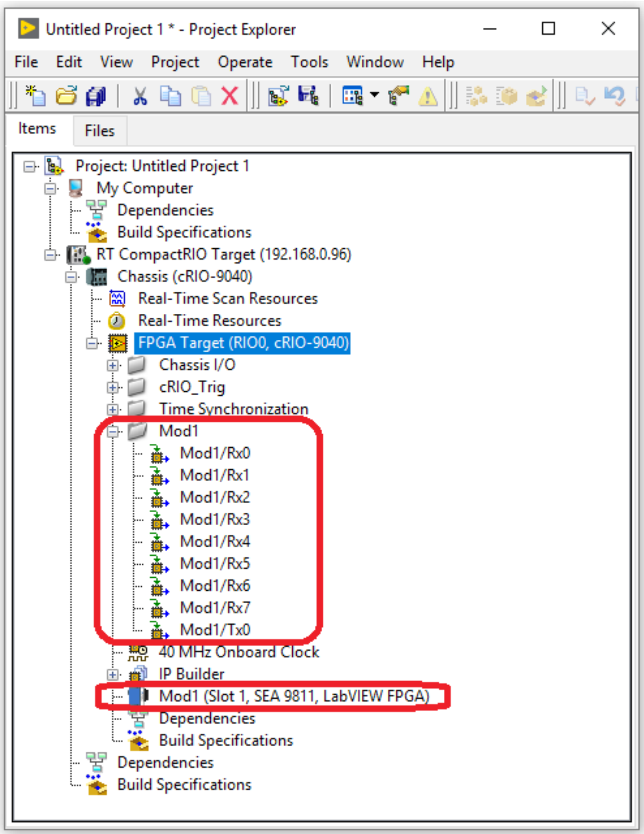


Fig. 3: Quick Start – Project explorer after completing the configuration

The project is now fully configured and the SEA 9811 resources can be used in the user FPGA application.

You can continue from here to learn how to implement a simple FPGA application using the SEA 9811 module.

11. Create a new FPGA VI in the project explorer window. For this right-click on the *FPGA Target (RIO0...)* and select *New -> VI*. Refer to figure 4 below:

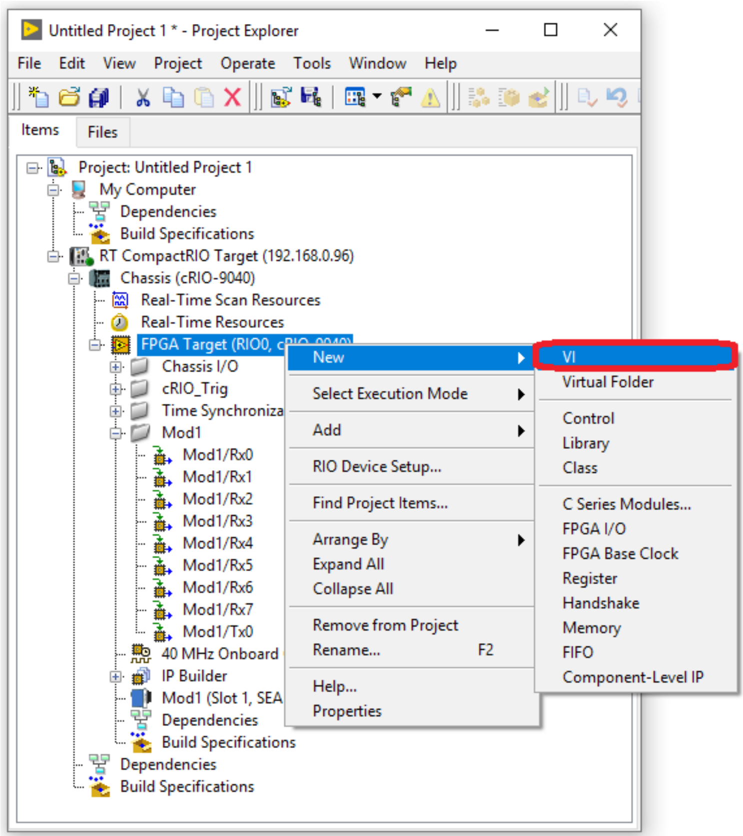


Fig. 4: Quick Start – Create FPGA VI

12. Open the block diagram of the VI just created and place a method node from the functions palette (*FPGA I/O -> I/O Method*) on the block diagram. Afterward select the item (SEA 9811 module) to execute the method on like shown below.

13. Finally select the method to be executed from the list of available methods for the selected item (SEA 9811 module). For this tutorial select the node *Rx Config Set* like shown below.

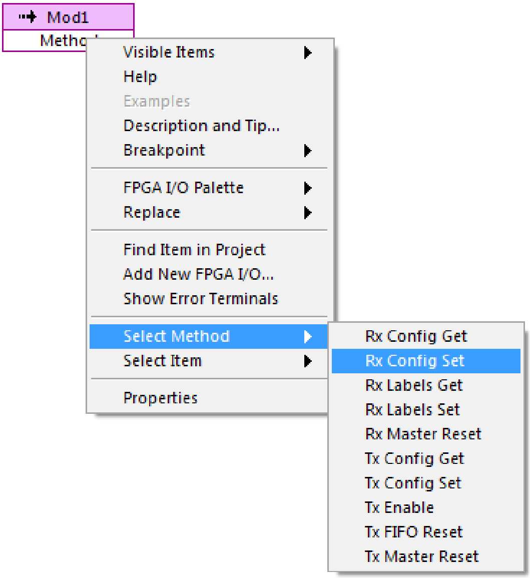


Fig. 6: Quick Start – Configure method node

14. Repeat steps 10 to 11 with an I/O node and select the first receiver channel *Mod1/Rx0*. Complete the example placing inputs/outputs for the nodes as well as a sequence to ensure that the configuration method node gets executed before the read operation. For the inputs use the values shown in the figure below.

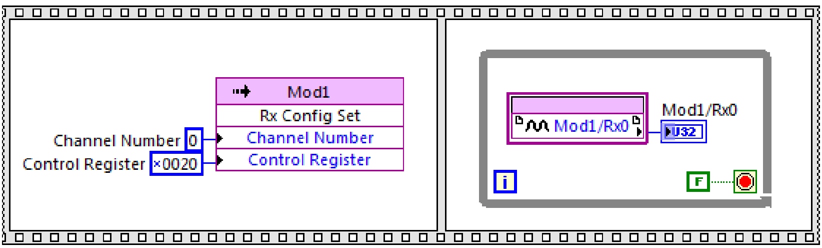


Fig. 7: Quick Start – Final block diagram

15. Save the created FPGA VI, create a build specification and compile it.
16. The *Mod1/Rx0* indicator will display the incoming data on channel 0 if any data was received.

    All functions are accessible via either IO, property or method node. For a complete list of function nodes please refer to section 7.2 Application Programming Interface (API).

6 Functional Overview

6.1 cRIO Platform

The SEA 9811 is a cRIO compatible module that can be used in a wide range of carriers from National Instruments. All CompactRIO systems with a programmable (FPGA) backplane are currently supported. SEA 9811 is not supported with systems without a FPGA programmable backplane like CompactDAQ.

6.2 ARINC Basics*

ARINC 429 is a technical standard for the avionics data bus used on aircrafts. It defines the physical and electrical interfaces of a two-wire data bus and a data protocol.

6.2.1 Messages

ARINC 429 is an application-specific standard for aircraft avionics. It's a self clocking self synchronizing data bus – that means that Tx and Rx are on separate ports - known as the Mark 33 *Digital Information Transfer System* (DITS). The physical connection wires are twisted pairs carrying balanced differential signaling.

Data words are 32 bits in length and most messages consist of a single data word. Messages are transmitted at either 12.5 or 100 kbit/s to other system elements that are monitoring the bus messages. The transmitter constantly transmits either 32-bit data words or the NULL state. A single wire pair is limited to one transmitter to up to 20 receivers.

6.2.2 ARINC Word Format

Each ARINC word is a 32-bit value that contains five fields:

- Bit 32 is the parity bit, and is used to verify that the word was not damaged or garbled during transmission. Every ARINC 429 channel typically uses *odd* parity - there must be an odd number of "1" bits in the word. This bit is set to 0 or 1 to ensure that the correct number of bits is set to 1 in the word.
- Bit 30 to 31 in the *Binary Coded Decimal* (BCD) or 29 to 31 in the binary data word is the *Sign/Status Matrix* (SSM) and often indicates whether the data in the word is valid.
  - *Normal Operation* (NO) - Indicates the data in this word is considered to be correct data.
  - *Functional Test* (FT) - Indicates that the data is being provided by a test source.
  - *Failure Warning* (FW) - Indicates a failure which causes the data to be suspect or missing.

** The information in this chapter base on the following article: Wikipedia - ARINC 429, 14 March 2011*

– *No Computed Data* (NCD) - Indicates that the data is missing or inaccurate for some reason other than a failure.

– The *SSM* can also indicate the sign (+/-) of the data or some information related to it like an orientation (North/South; East/West). In this case it is considered to be in *Normal Operation*.

SSM for BNR data		
SSM for BCD data		
Bit 31	Bit 30	Description
0	0	Plus, North, East, Right, To, Above
0	1	No Computed Data (NCD)
1	0	Functional Test (FT)
1	1	Minus, South, West, Left, From, Below

- Bit 11 to 29 contain the data. Bit-field, *Binary Coded Decimal* (BCD), and two's complement binary encoding (BNR) are common ARINC 429 data formats. Data formats can also be mixed.
- Bit 9 and 10 are *Source/Destination Identifiers* (SDI) and indicate the data destination or —more frequently— which subsystem transmitted the data.
- Bit 1 to 8 contain a label label words) expressed in octal, identifying the data type.

6.2.3 Labels

Label guidelines are provided as part of the ARINC 429 specification for various equipment types. Each aircraft will contain a number of different systems, such as flight management computers, inertial reference systems, air data computers, radar altimeters, radios or GPS sensors.

For each type of equipment, a set of standard parameters is defined, which is common across all manufacturers and models. For example, any air data computer will provide the barometric altitude of the aircraft as label 203. This allows interchangeability of parts, as all air data computers behave, for the most part, in the same way.

There is only a limited number of labels and so label 203 may have some completely different meaning if sent by a GPS sensor for example.

Also, as with any specification, each manufacturer has slight differences from the formal specification e.g. as by providing extra data above and beyond the specification, leaving out some data recommended by the specification.

6.3 Module Functionality

The SEA 9811 features the following functions:

- eight fully independent configurable receiver channels
- one configurable transmitter channel
- High- and low-speed transmission speed
- receiver label filtering with up to 16 labels per channel
- receiver self-test (loop back)
- status feedback

All functions are accessible as nodes within the FPGA programming level. Only one function can be executed at a time, which means that individual functions can be only executed subsequently.

The user has to ensure that two function are not executed at the same time, as this may lead to malfunction.

6.4 Power Up Behavior

At power up the module is initially configured, when any FPGA code starts to execute. The initial configuration disables all eight receiver channels setting each receiver channel control register to 0x0028. Refer to the section 7.3.1 Receiver Control Register (RCR) for details.

7 Programming

7.1 Examples

The SEA 9811 driver software is delivered with a set of examples that demonstrate a specific aspect of the module. Please refer to the examples to learn how to program the module and how to retrieve or send data.

The examples are available via the example finder. Use search keywords like ARINC or 9811 to find the related examples.

7.2 Application Programming Interface (API)

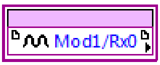
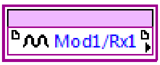
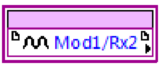
The Application Programming Interface (API) provides the user with access to the underlying ARINC chip set functionality. This interface exposes nodes that can be used within a LabVIEW FPGA application. The following different node types are available:

- IO nodes
- property nodes
- method nodes

7.2.1 IO Nodes

IO nodes deliver the measurement data from the hardware to the application (Rx IO node) or transports control data from the application to the hardware (Tx IO node).

The SEA 9811 provides eight Rx and one Tx IO nodes. Channels are always numbered beginning with 0.

Node	Direction	Description
	read only	Retrieves the channel 0 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN1A (+), IN1B (-).
	read only	Retrieves the channel 1 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN2A (+), IN2B (-).
	read only	Retrieves the channel 2 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN3A (+), IN3B (-).

Node	Direction	Description
Mod1/Rx3	read only	Retrieves the channel 3 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN4A (+), IN4B (-).
Mod1/Rx4	read only	Retrieves the channel 4 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN5A (+), IN5B (-).
Mod1/Rx5	read only	Retrieves the channel 5 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN6A (+), IN6B (-).
Mod1/Rx6	read only	Retrieves the channel 6 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN7A (+), IN7B (-).
Mod1/Rx7	read only	Retrieves the channel 7 data. If the respective FIFO is empty, a '0' is returned. Pins on the module: IN8A (+), IN8B (-).
Mod1/Tx0	write only	Write channel 0 data. Pins on the module: OA27 (+), OB27 (-) resp. OA37(+), OB37(-).

Tab. 1: SEA 9811 IO nodes

The data from the Rx IO nodes is read in a non-buffered manner (apart from the small module's hardware FIFO buffers with a depth of four words). The user is responsible to read the data fast enough and to optionally implement adequate mechanism to avoid data loss if this is critical. You can read the Rx status register (RSR) to find out if you are reading fast enough to prevent data loss.

7.2.2 Property Nodes

Property nodes can affect either the module itself, the receiver or the transmitter chips. Therefore, their names are pre-pended with a prefix: *Mod*, *Rx* and *Tx*. *Mod* are module wide properties; *Rx* properties apply only to the receiver; *Tx* properties affect only the transmitter.

Node	Direction	Description
Mod1 Mod ID	read only	Retrieves the module ID from the module. For SEA 9811 module it retrieves 0065 (0x0041).
Mod1 Mod SerialNumber	read only	Retrieves the serial number from the module. The serial number is a 8 digit number and can be either hexadecimal, decimal or in BCD code.
Mod1 Mod VendorID	read only	Retrieves the Vendor ID. This value identifies the manufacturer of the module. S.E.A. Modules always return 0x4711.
Mod1 Rx SimData HighSpeed	write only	Sets a ARINC Word that can be received by all receiver channels in high-speed mode. Every receiver channel needs to configured for loop back first. Refer to the receiver loop back example for details.
Mod1 Rx SimData LowSpeed	write only	The same meaning as above but for low-speed mode only.
Mod1 Rx Status	read only	Retrieves the status of the receiver (all channels). Refer to section 7.3.2 Receiver Status Register (RSR) for details. Rx FIFO status may be outdated quickly as data can be received any time.
Mod1 Tx Status	read only	Retrieves the status of the transmitter. Refer to section 7.3.5 Transmitter Status Register (TSR) for details.

Tab. 2: SEA 9811 property nodes

7.2.3 Method Nodes

Method nodes for the SEA 9811 only affect either the receiver or the transmitter. They are defined as follows:

Name	Direction	Description
Mod1 Rx Config Get Channel Number Control Register	read only	Retrieves the configuration (Control Register) of a specific receiver channel. For detail on the content refer to section 7.3.1 Receiver Control Register (RCR).
Mod1 Rx Config Set Channel Number Control Register	write only	Sets the configuration (Control Register) of a specified receiver channel. For detail on the content refer to section 7.3.1 Receiver Control Register (RCR).

Name	Direction	Description
		The channel labels and FIFO are resetted.
Mod1 Rx Labels Get Channel Number Label Filter	read only	Retrieves the current set of filter labels for a specific receiver channel.
Mod1 Rx Labels Set Channel Number Label Filter	write only	Send a set of filter labels to a specific receiver channel.
Mod1 Rx Master Reset	write only	Resets the FIFO buffers and the labels of all receiver channels. The configuration registers remain unchanged.
Mod1 Tx Config Get Control Register	read only	Retrieves the configuration (Control Register) of the transmitter. For detail on the content refer to section 7.3.4 Transmitter Control Register (TCR).
Mod1 Tx Config Set Control Register	write only	Sends configuration (Control Register) to the transmitter. For detail on the content refer to section 7.3.4 Transmitter Control Register (TCR).
Mod1 Tx Enable	write only	Prompts the transmitter to send out data. The data must have been previously loaded into the transmitter FIFO using the Tx IO node. This method is only necessary if configuration bit 13 (TCR13) is 0.
Mod1 Tx FIFO Reset	write only	Empties the transmitter FIFO.
Mod1 Tx Master Reset	write only	Terminates the data transmission immediately and empties the transmitter FIFO.

Tab. 3: SEA 9811 method nodes

7.3 Data Formats

The data exchanged between the SEA 9811 and the application software is organized in words with different meanings. There are control and status registers to define receiver and transmitter behavior as well as the actual ARINC data words.

7.3.1 Receiver Control Register (RCR)

The are eight Receiver Control Registers; one for each receiver channel. The RCR is a 16 Bit Word with the following format:

Bit	Function	Description
RCR0 (LSB)	Data Rate	0 = Receiver in High Speed Mode (100 kbit/sec) 1 = Receiver in Low Speed Mode (12.5 kbit/sec)
RCR1	RFLAG	Not used.
RCR2	Label Filter Enable	0 = Receiver Label Filtering disabled (all labels are valid) 1 = Receiver Label Filtering enabled
RCR3	Reset	0 = Receiver in normal operation 1 = Receiver channel disabled. Clears channel logic and FIFO. Should be applied if a channel is not used. This is also the default setting for all channels at start up.
RCR4	Parity Check Enable	0 = Receiver Parity check disabled 1 = Receiver Odd parity check enabled
RCR5	Self-Test Enable	0 = Receiver Self-Test enabled (loop back operation) 1 = Receiver Self-Test disabled (normal operation)
RCR6	Decoder Enable	0 = Receiver Decoder disabled 1 = Receiver decoding enabled. ARINC Word bits 10 and 9 must match RCR7 and RCR8.
RCR7	—	If RCR6=1 the ARINC Word Bit 10 must match this bit
RCR8	—	If RCR6=1 the ARINC Word Bit 9 must match this bit
RCR9	ARINC Label Bit Order	0 = Receiver Label Bit Order Reversed. See Tab. 6: ARINC Word. 1 = Receiver Label Bit order same as received.
RCR10-15	Not Used	Control Register read operation returns always "0"

Tab. 4: Receiver Control Register

7.3.2 Receiver Status Register (RSR)

Every channel has a input FIFO with a depth of 4. That means the last 4 received data packages are available. The RSR delivers information about these FIFOS. The format of RSR is as follows:

Bit	Function	Description
RSR0 (LSB)	Channel 1 FIFO Empty	0 = Receiver Channel 0 FIFO contains valid data. 1 = Receiver Channel 0 FIFO empty; All data has been read.
RSR1	Channel 2 FIFO Empty	0 = Receiver Channel 1 FIFO contains valid data. 1 = Receiver Channel 1 FIFO empty; All data has been read.
RSR2	Channel 3 FIFO Empty	0 = Receiver Channel 2 FIFO contains valid data. 1 = Receiver Channel 2 FIFO empty; All data has been read.
RSR3	Channel 4 FIFO Empty	0 = Receiver Channel 3 FIFO contains valid data. 1 = Receiver Channel 3 FIFO empty; All data has been read.
RSR4	Channel 5 FIFO Empty	0 = Receiver Channel 4 FIFO contains valid data. 1 = Receiver Channel 4 FIFO empty; All data has been read.
RSR5	Channel 6 FIFO Empty	0 = Receiver Channel 5 FIFO contains valid data. 1 = Receiver Channel 5 FIFO empty; All data has been read.
RSR6	Channel 7 FIFO Empty	0 = Receiver Channel 6 FIFO contains valid data. 1 = Receiver Channel 6 FIFO empty; All data has been read.
RSR7	Channel 8 FIFO Empty	0 = Receiver Channel 7 FIFO contains valid data. 1 = Receiver Channel 7 FIFO empty; All data has been read.
RSR8..15	Channel 1..8 FIFO Full	0 = Receiver Channel 0 FIFO is not full. 1 = Receiver Channel 0 FIFO is full.

Tab. 5: Receiver Status Register

7.3.3 ARINC Word

The ARINC Word transports the data between the transmitter and the receiver. It is a 32 bit long word with the following format:

Bit No.	RCR9 = 0 (when receiving) TCR11 = 0 (when transmitting)		RCR9 = 1 TCR119 = 1	
	ARINC Word bit	**Function**	**ARINC Word bit**	**Function**
0	31	Parity / Data	31	Parity /Data
1-21	30-10	Data	30-10	Data
22	9	SDI	9	SDI
23	8	SDI	8	SDI
24	7	Label (MSB)	0	Label (LSB)
25	6	Label	1	Label
26	5	Label	2	Label
27	4	Label	3	Label

Bit No.	RCR9 = 0 (when receiving) TCR11 = 0 (when transmitting)		RCR9 = 1 TCR119 = 1	
28	3	Label	4	Label
29	2	Label	5	Label
30	1	Label	6	Label
31	0	Label (LSB)	7	Label (MSB)

Tab. 6: ARINC Word

7.3.4 Transmitter Control Register (TCR)

The transmitter can be configured via the Transmitter Control Register. The TCR is a 16 Bit Word with the following format:

Bit	Function	Description
TCR0 (LSB)	—	Not used
TCR1	—	Reserved. Must always be 0.
TCR2	—	Not used
TCR3	Parity Bit Enable	0 = transmitter 32nd bit is data. 1 = transmitter 32nd bit is parity.
TCR4	—	Not used
TCR5	—	Not used
TCR6	—	Not used
TCR7	—	Not used
TCR8	—	Not used
TCR9	Parity	0 = transmitter 32nd bit id Odd parity. 1 = transmitter 32nd bit is Even parity.
TCR10	Data Rate	0 = High Speed Mode (100 kbit/sec) 1 = Low Speed Mode (12.5 kbit/sec)
TCR11	ARINC Label Bit Order	0 = Label Bit order reversed. 1 = Label bit order same as transmitted.
TCR12	Disable Line Driver	0 = Line Driver enabled. 1 = Line driver disabled.

Bit	Function	Description
TCR13	Transmission Enable Mode	0 = Start transmission by command (Tx Enable, refer to Tab. 3: SEA 9811 method nodes) 1 = Transmit immediately when data is available in transmitter FIFO.
TCR14	—	Reserved.
TCR15	—	Not used

Tab. 7: Transmitter Control Register

7.3.5 Transmitter Status Register (TSR)

The transmitter Status Register is a 8 Bit Word and returns the data transmission information. The transmitter has an output FIFO with a depth of 32. The TSR delivers information about this FIFO. The format of TSR is as follows:

Bit	Function	Description
TSR0 (LSB)	—	Not used
TSR1	—	Not used
TSR2	—	Not used
TSR3	FIFO Empty	0 = Transmit FIFO is not empty. 1 = Transmit FIFO is empty. All data in FIFO has been sent.
TSR4	FIFO Half Full	0 = Transmit FIFO contains less than 16 words. 1 = Transmit FIFO contains at least 16 words.
TSR5	FIFO Full	0 = Transmit FIFO is not full (less than 32 words). 1 = Transmit FIFO is full.
TSR6	—	Reserved; is always "0"
TSR7	—	Reserved; is always "0".

Tab. 8: Transmitter Status Register

7.4 Error Codes

In case of unexpected behavior the driver software returns an error within the error cluster. The option S*how Error Terminals* of a node must be enabled in this case. Following errors are possible:

Error Code	Description
65536	No module or invalid module type found. Please insert SEA 9510 module into the correct slot. If a SEA 9510 module is recognized as invalid, please update the module's firmware as described in chapter 8 .
65537	Incorrect module type found. Only SEA 9811 modules will work with this driver version. Please check if all modules are inserted into the right slot.

Tab. 9: Error Codes

7.5 FPGA Usage and Optimization

The SEA 9811 driver software consumes approx. 30% of available FPGA space, when using the NI 9104 backplane. Note that the FPGA space consumption for the driver code is not linear in any way. It varies with the chassis types and number of total modules installed.

However, the current driver software architecture offers a way that can help to reduce the FPGA space further if FPGA space is precious. Within the SEA 9811 driver core certain functional blocks can be disabled using *Conditional Disable Symbols*. A disabled functional block is not compiled into the bitfile and therefore does not consume FPGA space. When using *Conditional Disable Symbols* some API functions become non-executable and must not be used to prevent unexpected behavior. Please refer to the table below for details on using this optimization feature:

Conditional Disable Symbol	Value Range	Description
EEPROM	OFF ON (default)	OFF disables the following nodes: • Mod ID • Mod SerialNumber • Mod VendorID
LABEL	OFF ON (default)	OFF disables the following nodes: • Rx Labels Get • Rx Labels Set

Conditional Disable Symbol	Value Range	Description
TX	OFF ON (default)	OFF disables all Tx related nodes: • Tx Config Get • Tx Config Set • Tx Status • Tx Enable • Tx FIFO Reset • Tx Master Reset

Tab. 10: Conditional Disable Symbols

The *Conditional Disable Symbols* can be defined in the project property dialog box like shown in Fig. 8: Project Properties – Conditional Disable Symbols.

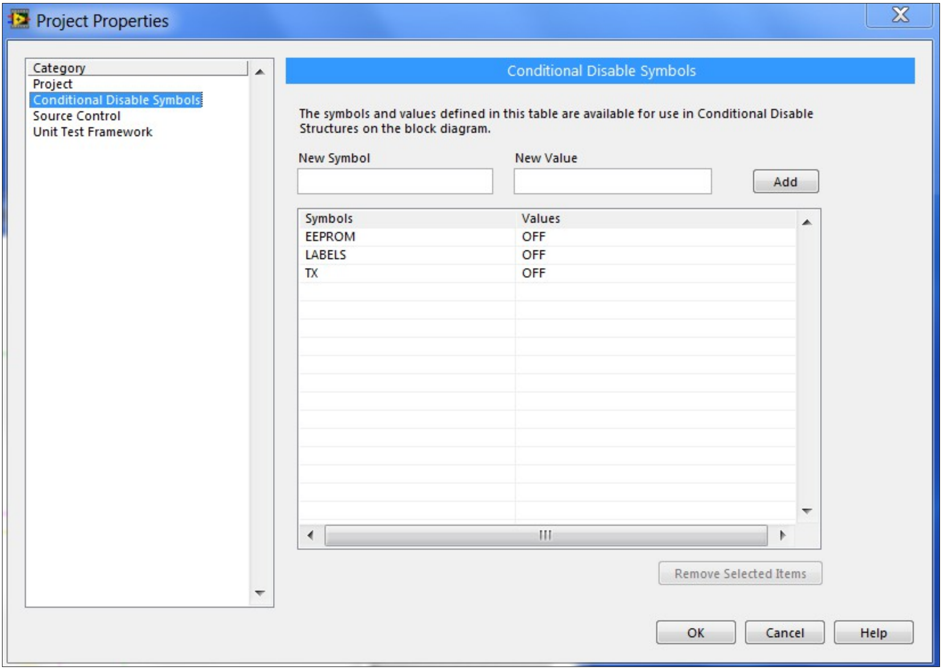


Fig. 8: Project Properties – Conditional Disable Symbols

This dialog box is accessible by right-click on the project root item "Project: *‹project n me›*" in the LabVIEW project explorer. If no conditional disable symbol are defined the associated functional blocks are enabled by default.

8 Module Updater

The SEA 9811 module must be correctly updated before it can be used with the new MDK2 series driver software. If unsure which module version you have you might want to start the module updater to check if an update is required.

You can start the module updater either after installation directly from the Welcome screen pressing *Update Module* button or later from LabVIEW menu bar: *Help -> SEA -> SEA 9811 -> Update Module...*

Please read carefully the instructions below. The main VI is **MDK2-MU_Host.vi** residing in the **My Computer** execution target.

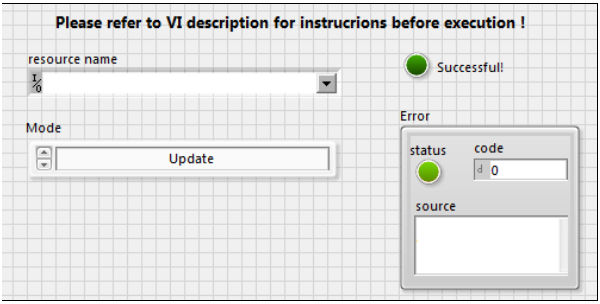


Fig. 9: Module Updater – MDK2_MU_Host

AUDIENCE:

This update is only required for modules that are shipped before November 2012 AND use the following driver software:

- ARINC: 2.0 or higher

Update the module eeprom only if ALL conditions apply. If any vagueness remains please contact S.E.A.: techsupport@sea-gmbh.com for clarification.

PREREQUISITES:

1. CompactRIO System (Real-Time controller, FPGA chassis) with LabVIEW 2011
2. S.E.A. cRIO module

OPERATION DESCRIPTION:

Two modes are provided:

1. **Update:**
   Reads the current module eeprom and checks if update is required. In no, nothing is done. If yes the current eeprom content is written to the file **MDK2-MU_Eeprom_Before.txt** (backup before update). Afterwards the eeprom is updated and written back to the module. Additionally the new eeprom content is written to another file **MDK2-MU_Eeprom_After.txt** (backup after update).

2. **Restore From Backup File:**
   Reads the backup file **MDK2-MU_Eeprom_Before.txt** and writes its content to the module. This is only performed, when a valid S.E.A. CRIO module is found.

OPERATION INSTRUCTION:

1. Insert the Module that should be updated in **Slot 1** of your CompactRIO system
2. Detect your CompactRIO system, **DO NOT** discover the cRIO (C Series) modules
3. In the project explorer window copy the following FPGA Target items from the default CompactRIO system (0.0.0.0) to the same project item of your discovered CompactRIO system:
   - MDK2-MU_FPGA.vi
   - Mod1 (Slot1, MDK2-MU)
4. Create Build Specification for **MDK2-MU_FPGA.vi** and compile it.
5. After the FPGA compilation has finished open the **MDK2-MU_Host.vi** in **My Computer** target.
6. Relink the new compiled FPGA VI in the FPGA Open VI Reference node.
7. Execute the **MDK2-MU_Host.vi**
8. Upon a successful execution the Module is updated. Please save two files for backup purposes:
   - ..\Data\MDK2-MU_Eeprom_Before.txt – this is the backup before update
   - ..\Data\MDK-MU_Eeprom_After.txt – this is the backup after update

9 Trouble Shooting

9.1 Common Problems and Solutions

The following hints shall help you to determine if the module and/or software behave correctly and how to identify module failures.

#	Problem	How to solve
1	The module is not detected within LabVIEW	Make sure you have the system powered and the module has the correct firmware. Only new module version are detected with the module detector. You can start the firmware update tool to check the modules firmware setup.
2	No labels are detected or retrieved	Check the receive status register if there are labels currently available in the input buffer. If the status register indicates data you should be able to retrieve the labels. If the register indicates the input buffer is empty, start the loop back test and see if you receive label data. If you receive any label data, probably your external wiring is not correct. Check channel wiring. Make sure you have the correct channel allocation. ARINC channels start with number 0 and end with channel number 7.
3	Loopback function does not work	Module is possibly damaged. Retry after power cycle of the module. If the problem still exists contact the supplier of the module.
4	Labels are not send	If you try to receive the labels with the built in ARINC receiver,make sure you have the correct wiring. See hardware manual for channel and connector allocation. Use a ARINC crossover cable to send an receive labels with the SEA 9811.

9.2 ARINC Self-Test Adapter

For functional testing a special adapter that connects the transmitter with all receiver channels may be useful. The principle of this circuit is to connect the positive transmitter output (OA27) with all positive receiver inputs (INxA) and the negative transmitter output (OB27) with all negative receiver inputs (INxB). Please refer to the figure below on how to build such an adapter.

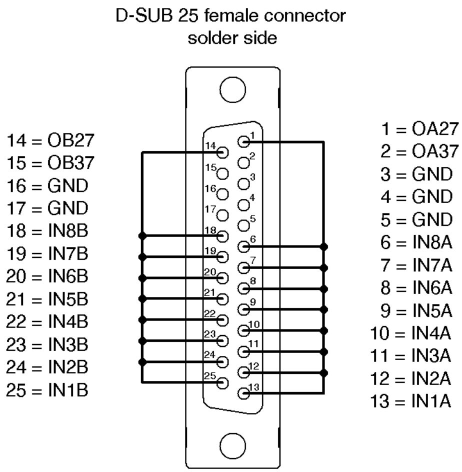


Fig. 10: ARINC Self-Test Adapter

This adapter can be directly plugged in the SEA 9811 module. The transmitted data is received at all receiver channels.

Please also note that the *ReceiverAndTransmitter* example included with the driver software requires such a circuit to demonstrate the operation.

A Figures

B Tables