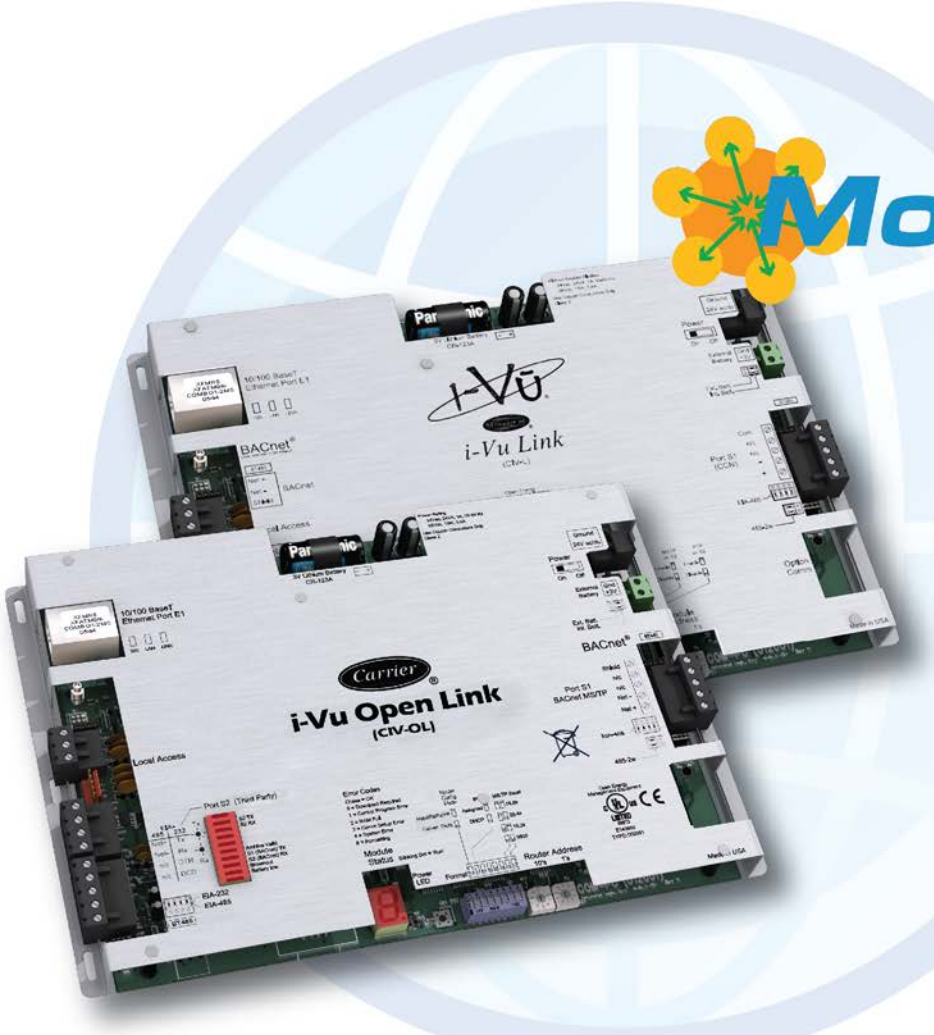
Modbus Integration Guide
i-Vu® Link and i-Vu® Open Link



CARRIER CORPORATION ©2015
A member of the United Technologies Corporation family · Stock symbol UTX · Catalog No. 11-808-515-01 · 7/7/2015

Verify that you have the most current version of this document from **www.hvacpartners.com** or your local Carrier office.

Important changes are listed in **Document revision history** at the end of this document.

CARRIER CORPORATION © 2015. All rights reserved throughout the world. i-Vu is a registered trademark of Carrier Corporation. BACnet is a registered trademark of ASHRAE. All other trademarks are the property of their respective owners.

Contents

Overview **1**

Controller as a master device on EIA-232/485 ..... **2**

Before-you-begin checklist **2**

The integration process **3**

1 Create a control program in the Snap application **3**

Formatting a Modbus address 3

Editing a microblock address 5

2 Download the Modbus driver and control programs **6**

3 Connect the Carrier controller to the third-party device **6**

Wiring specifications 6

To wire a third-party device 7

4 Set up the driver properties for Modbus EIA-232/485 ..... **7**

5 Verify the controller is set up correctly **8**

To capture communication using PuTTY 9

Controller as a slave device on EIA-232/485 ..... **12**

Before-you-begin checklist **12**

The integration process **13**

1 Create a control program in the Snap application **13**

Formatting a Modbus address 13

Editing a microblock address 15

2 Download the Modbus driver and control programs **15**

3 Connect the Carrier controller to the third-party device **16**

Wiring specifications 16

To wire a third-party device 16

4 Set up the driver properties for Modbus EIA-232/485 ..... **17**

5 Verify the controller is set up correctly **18**

To capture communication using PuTTY 19

Controller as a client device on Ethernet **21**

Before-you-begin checklist **21**

The integration process **22**

1 Create a control program in the Snap application **22**

Formatting a Modbus address 22

Editing a microblock address 24

2 Download the Modbus driver and control programs **25**

3 Connect the Carrier controller to the third-party device **25**

4 Set up the driver properties for Modbus TCP/IP **26**

5 Verify the controller is set up correctly **26**

To capture communication using Wireshark 28

To capture communication using PuTTY 29

Controller as a server device on Ethernet **31**

Before-you-begin checklist **31**

The integration process **32**

1 Create a control program in Snap **32**

Formatting a Modbus address 32

Editing a microblock address 34

2 Download the Modbus driver and control programs **34**

3 Connect the Carrier controller to the third-party device **35**

4 Set up the driver properties for Modbus TCP/IP **35**

5 Verify the controller is set up correctly **36**

To capture communication using Wireshark 37

To capture communication using PuTTY 38

Modbus Troubleshooting **40**
If the Carrier controller is a Serial Master or Ethernet Client **40**
Register addresses 40
Errors 41
If the Carrier controller is a Serial Slave or Ethernet Server **45**
Register addresses 45
Errors 45
Appendix A - Modbus Protocol Conformance Statement **46**
Document revision history **48**

Overview

A controller from Carrier can be used to integrate Modbus device(s) with your i-Vu® system. The controller can serve as a master or slave on a serial network, or a client or server on a TCP/IP network.

This document is divided into 4 sections that provide integration instructions for the 4 scenarios listed in the first two columns below. Determine how your Carrier controller will be used, then go to the appropriate page for the instructions.

If the Carrier controller is to be a Modbus...	**...on this media type...**	**...you can use this port...**	**See...**
Master	EIA-485, 2 wire EIA-232	S2 S2	*Controller as a master device on EIA-232/485* (page 2)
Slave	EIA-485, 2 wire EIA-232	S2 S2	*Controller as a slave device on EIA-232/485* (page 12)
Client	Ethernet	E1	*Controller as a client device on Ethernet* (page 21)
Server	Ethernet	E1	*Controller as a server device on Ethernet* (page 31)

NOTES

- The Modbus driver supports Modbus devices connected to port S2, and E1 simultaneously on the Carrier controller. The third-party point count for the router is the total of the 2 ports.
- The 2 ports can simultaneously have different functionality. For example, Port S2 could be a serial slave, and Port E1 an Ethernet client.

CARRIER CORPORATION ©2015
All rights reserved

Controller as a master device on EIA-232/485

NOTE Modbus supports a single master on each network.

Carrier		
Controllers	i-Vu® Link	i-Vu® Open Link
Module Driver	drv_ivulink_modbus_6-00-082* drv_ivuopenlink_std_6-00-082*	
Read/write capability	Can read from and write to the third-party equipment	
Third party		
Supported equipment	Any device that supports the Modbus (Modicon) protocol	
Network media type	EIA-232, EIA-485	
Quantity of Modbus slave devices you can physically connect to Port S2 on the Carrier master controller	For EIA-232:	1 Modbus slave
	For EIA-485:	Up to 254 Modbus slaves **NOTE** You may reach the point limitation of the Carrier controller before connecting 254 Modbus slaves.

***You can download the latest drivers from the *Carrier Control Systems Support Site http://www.hvacpartners.com/*. Check the latest bulletin releases for new or updated drivers.

Before-you-begin checklist

You need the following items, information, and skills for the integration process:

- ☐ A points list for each Modbus device that includes register addresses and read/write capabilities. Points lists are usually available from the third-party manufacturer's representative or website.
- ☐ The addresses of the Modbus devices
- ☐ The network media type required by the Modbus device: EIA-232 or EIA-485
- ☐ A port pinout/configuration of the Modbus device. Pinouts are usually available from the third-party manufacturer's representative or website.
- ☐ The Modbus network's baud rate, data bits, parity, and stop bits

CARRIER CORPORATION ©2015
All rights reserved

- ☐ Verification that all communication properties have been set on the Modbus devices
- ☐ Verification of communications through the port that the Carrier controller will connect to
- ☐ Experience creating control programs in Snap
- ☐ Experience installing, wiring, setting up, and downloading to the Carrier controller

The integration process

Follow the steps in this section to integrate one or more third-party Modbus devices into an i-Vu® system using an i-Vu Link or i-Vu Open Link. To install and network a Link, see the Installation and Start-up Guide.

1 Create a control program in the Snap application

When you create your control program, use a Network I/O microblock for each third-party point.

TIP Verify the third-party manufacturer's register addressing pattern by establishing communication with a few points whose values you can physically manipulate before you spend time addressing the remaining integration points.

Formatting a Modbus address

Use the information below to format a valid address in each microblock that you use to read or write to a third-party point.

CAUTION!
When integrating third-party devices into a i-Vu® system, most communication problems are caused by incorrect data or typing errors in the microblock's **Address** field.

Address format:

modbus://register type/modbus register address/device address*

Defined by Carrier. See table on next page.	Provided in third-party points list. See table below.	Set on third-party device (1-247)

Example: modbus://UINT/40128/26

NOTE To send a broadcast to all Modbus devices on the port specified by the protocol signature, use 0 for the device address in a network output microblock.

CARRIER CORPORATION ©2015
All rights reserved

Register type/Register address

To...	this kind of value...	use this microblock...	with this register type...	and a Modbus register address in this range... [1]
Read	0 to 65,535 Unsigned 16-bit integer Input register (**3**XXXXX)	ANI	Uint (not Unit)	**3**00001–**3**65535
	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANI	Uint (not Unit)	**4**00001–**4**65535
	0 to 4,294,967,296 Unsigned, 32-bit (long) integer Input register (**3**XXXXX)	ANI	Uint32	**3**00001–**3**65535
	0 to 4,294,967,296 Unsigned, 32-bit (long) integer Holding register (**4**XXXXX)	ANI	Uint32	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Input register (**3**XXXXX)	ANI	Sint	**3**00001–**3**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANI	Sint	**4**00001–**4**65535
	-2,147,483,648 to 2,147,483,647 Signed, 32-bit (long) integer Input register (**3**XXXXX)	ANI	Sint32	**3**00001–**3**65535
	-2,147,483,648 to 2,147,483,647 Signed, 32-bit (long) integer Holding register (**4**XXXXX)	ANI	Sint32	**4**00001–**4**65535
	Value with decimal point Input register (**3**XXXXX)	ANI	Float	**3**00001–**3**65535 [2]
	Value with decimal point Holding register (**4**XXXXX)	ANI	Float	**4**00001–**4**65535 [2]
	0 or 1 Coil / Discrete (binary) output	BNI (or ANI)	Do	1–65535 [3]
	0 or 1 Discrete (binary) input (**1**XXXXX)	BNI (or ANI)	Di	**1**00001–**1**65535
	0 or 1 Input register (**3**XXXXX)	BNI	BIT*n* (where *n* is a value 0-15 defined in points list)	**3**00001–**3**65535
	0 or 1 Holding register (**4**XXXXX)	BNI	BIT*n* (where *n* is a value 0-15 defined in points list)	**4**00001–**4**65535

CARRIER CORPORATION ©2015
All rights reserved

To...	this kind of value...	use this microblock...	with this register type...	and a Modbus register address in this range... [1]
Write	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANO	Uint (not Unit)	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANO	Sint	**4**00001–**4**65535
	Value with decimal point Holding register (**4**XXXXX)	ANO	Float	**4**00001–**4**65535 [2]
	0 or 1 Coil Discrete (binary) output	BNO (or ANO)	Do	1–65535 [3]
	0 or 1 Holding register (**4**XXXXX)	BNO (or ANO)	BIT*n* (where *n* is a value 0-15 defined in points list)	**4**00001–**4**65535

[1] The Modbus register address (sometimes called register, address, code, or parameter) must be a decimal value. If the register addresses in the points list include the letters A-F, use a scientific calculator to convert these hexadecimal values to decimal values.

If the register addresses in the points list are 5-digit numbers in the range **#**0001-**#**9999 (where **#** = the prefix **1**, **3**, or **4**), you may use the addresses as they are, or change them to fit the formats in the table above by shifting the prefix one digit to the left. For example, if you see the address **3**8129, you can also use **3**08129.

[2] Each Float uses 2 consecutive Modbus register addresses. Use the lower number in the microblock address.

[3] Enter the number without any leading zeros. For example, enter 125, not 000125.

See *Modbus Troubleshooting* (page 40) for more information on register addresses.

Editing a microblock address

You can edit a microblock address in the following places:

- In the Snap Property Editor
- In the i-Vu® interface on the microblock's **Properties** page > **Details** tab
- In the i-Vu® interface on the control program's **Properties** page > **Network Points** tab

CARRIER CORPORATION ©2015
All rights reserved

2 Download the Modbus driver and control programs

If you do not already have the latest SAL library or drivers, download it from *Carrier Control Systems Support Site http://www.hvacpartners.com/* and save it to your computer.

To apply the SAL in the i-Vu interface:

1 Click **Main Menu**, then select **System Options** tree > **Update** tab.

NOTE Expand **Current Libraries (.sal)** to see the current SAL libraries and their revision. Compare them to what you downloaded and determine if any of them need updating.

2 Click Update Library and browse to the updated .sal file that you have saved on your computer, select the file, and click **Open**.

3 Click **Continue**.

4 When process is complete, the message appears **File added successfully**.

5 Click **Close**.

NOTE These changes are not applied until you have updated routers and controllers.

1 On the i-Vu® navigation tree, Right-click the router that you wish to update and click **Driver Properties**.

2 Select **Properties** page > **Update** tab.

3 If the database contains 2 or more routers, you must check **Change for all control programs of this type** in the **Controller** section.

4 Click **Update**. A message appears **Changes the driver and screen file to use the current library version. Continue?**

NOTE If more than one router exists, the additional routers are listed below the **Update** button.

5 Click **OK.**

6 Click **Accept.**

7 Continue to update any other necessary devices.

3 Connect the Carrier controller to the third-party device

Wiring specifications

For...	Use...	Maximum Length
EIA-232	18–28 AWG; twisted pair preferable	50 feet (15.24 meters)
EIA-485, 2-wire	18–24 AWG twisted, shielded pair [1]	3000 feet (914.4 meters)

[1] Shielding provides noise immunity in an electrically noisy environment. Acceptable wiring types, in order of greatest to least noise immunity, are:
- 2-pair, each pair individually shielded
- 2-pair, single overall cable shield

CARRIER CORPORATION ©2015
All rights reserved

To wire a third-party device

1. Turn off the i-Vu® Link/Open Link's power.
2. Check the communications wiring for shorts and grounds.
3. Wire the i-Vu® Link/Open Link's Port S2 to the third-party device, then set the S2 jumper. See table and notes below.
4. Turn on the i-Vu® Link/Open Link's power.

For...	Use i-Vu® Link/Open Link port...	Wire Carrier terminal...	...to third-party device terminal	Set the port's jumper(s) on i-Vu® Link/Open Link
EIA-232	S2	TX Rx Signal Ground	Rx Tx Gnd	EIA-232
EIA-485, 2-wire	S2	Net+ Net-	+ -	EIA-485 2-wire

NOTES

- If you cannot determine the media type or connections of the third-party device, contact your third-party representative.
- Use the same polarity throughout the network segment.
- Repeaters are required for more than 31 devices. See your third-party device manufacturer's recommendations.
- To reduce communication and data errors, terminate each end of an EIA-485 network with a resistor whose value equals the network's characteristic impedance. Some third-party manufacturers provide a built-in resistor that you enable or disable with a jumper. Make sure that only devices at the end of a network have termination enabled.

  EXAMPLE If an EIA-485 2-wire network's characteristic impedance is 120 Ohms, terminate one pair by placing a 120 Ohm resistor across the **Net+** and **NET-** connectors of the i-Vu® Link/Open Link. Terminate the other pair by placing a 120 Ohm resistor across the **+** and **-** connectors of the furthest third-party controller.
- A solid receive light on the i-Vu® Link/Open Link indicates a wiring or polarity problem.

4 Set up the driver properties for Modbus EIA-232/485

1. On the i-Vu® navigation tree, right-click the i-Vu® Link or i-Vu® Open Link and select **Driver Properties**.
2. Expand **Protocols**, then select **Modbus**.
3. Under **Port Configuration**, check **Enable** under **Port S2**.
4. Set the remaining fields under **Port Configuration** to match the third-party device.
5. Under **Protocol Configuration**, verify that **Is this device a Master** is set to **Yes**.
6. Set the remaining fields using information from the third-party manufacturer's representative. Check **Details** for help.

CARRIER CORPORATION ©2015
All rights reserved

7 Click **Apply**.

8 Select **Protocols** in the navigation tree.

9 In the **Protocol Status** table, verify that Modbus shows **Running** on Port S2

If the status shows **Not Running** or the wrong port, verify that:

- You selected the correct port in step 3 above.
- You set the controller's DIP switches correctly.
- The control program has at least one valid Modbus address. The driver must detect at least one valid address to initiate Modbus communications.

5 Verify the controller is set up correctly

1 On the i-Vu® navigation tree, select the controller.

2 Select the **Properties** page > **Network Points** tab.

If...	Then...
You see the point value you expect with no errors in the **Error** column	You have successfully established communication with the third-party device.
All points show question marks instead of values	The i-Vu® application is not communicating with the Carrier controller or the control program. Troubleshoot the controller's communications. See the controller's Installation and Start-up Guide.
Some points show question marks instead of values	You may have exceeded the third-party points available in the controller. If so, do one of the following: • Use a controller that provides more third-party points. • Split the points between two control programs used in separate controllers. To determine the number of third-party points used in a controller: On the i-Vu® navigation tree, right-click on the controller and select **Driver Properties** > **Driver** > **Properties** > **Settings** tab, and then scroll to **Network Microblocks**. **Number of integration points requested** and **Number of integration points active** show how many third-party Network I/O microblocks the controller is using. These counts will differ if you exceed the controller's integration point limit. For example, if your i-Vu® Link/Open Link's control program includes 27 third-party points, your **Integration points requested** will be 27 and your **Integration points active** will be 25.

CARRIER CORPORATION ©2015
All rights reserved

If...	Then...
The point name is red	Select the controller on the i-Vu® navigation tree. Select **Properties** > **Network Points** tab. Look in the **Error** column for one of the following error codes and descriptions. • **1 - Communications Disabled for this Microblock** Enable the microblock's **Communications Enabled** field on the **Network Points** tab, on the microblock's **Properties** page > **Details** tab, or in the Snap application. • **3 - Address Error - Unknown Protocol Specified** Select the correct port on the Modbus driver page, verify that **Address** in the microblock is correct, or set the DIP switches correctly on the controller. • All other errors On the i-Vu® navigation tree, right-click the controller and select **Driver Properties**, Carrier controller's driver, then select **Protocols** > **Modbus Error Definitions**. Find the error and its possible solution.
A value is incorrect	Verify that: • The **Address** in the microblock is correct. • The retrieved value is scaled properly, if necessary. For example, scaled from Celsius to Fahrenheit. Refer to the third-party manufacturer's documentation or the controller's Installation and Start-up Guide for scaling information.

If the above solutions do not resolve the problem, gather the following information for technical support:

- A diagnostic capture. See next topic.
- A screenshot of the **Driver Properties** > **Protocols** > **Properties** page and the **Driver** > **Protocols** > **Modbus** > **Properties** page
- A screenshot of the **Properties** page > **Network Points** tab showing addresses and errors
- Technical instructions for the third-party device, if available

To capture communication using PuTTY

Use PuTTY, a free open source terminal emulation program that works with all Windows operating systems, to capture communication between the controller and the Modbus device into a text file.

PREREQUISITES

- Download and install PuTTY from the *PuTTY website* (*http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html*).
- Get the Carrier controller's IP address.
- Get the longest microblock refresh time in the control program that has the error you are troubleshooting. In the i-Vu® interface, view the control program's **Properties** page > **Network Points** tab to see all the refresh times.

CARRIER CORPORATION ©2015
All rights reserved

1. Connect your computer's Ethernet port to the controller's Ethernet port using one of the following:
   - A CAT5 or higher Ethernet crossover cable
   - A hub and a CAT5 or higher Ethernet straight-through cable
2. Ping the Carrier controller to verify communications between the controller and the computer you are using to get this capture.
3. On the i-Vu® navigation tree, right-click the controller, select **Driver Properties**, then select **Protocols**.
4. Click **Properties**, select **Enable Telnet diagnostics**, then click **Accept**.
5. Start PuTTY.
6. Enter the controller's IP address.
7. Select **Telnet** as the **Connection Type**.
8. Click the **Logging** option in the upper left corner of the left pane.
9. Select **Printable output**.
10. Select a location for the capture text file.
11. Click **Open** to start the session.
12. After **Login:>**, type: `diagport`
    Press **Enter**.
    - To capture data receipts, after **diagport>**, type: `modbus rx`

    Press **Enter**.
    - To capture data transmissions, after **diagport>**, type: `modbus tx`

    Press **Enter**.
    - To capture more detailed diagnostic messages, after **diagport>**, type: `modbus vmsg`

    Press **Enter**.
    - To capture error messages, after **diagport>**, type: `modbus emsg`

    Press **Enter**.
13. Verify the displayed text shows:

```
modbus reporting level status:
      rx on
      tx on
      flush off
      fc1 off
      fc2 off
      vmsg on
      emsg on
          off
```

    If `rx`, `tx`, `vmsg`, or `emsg` show `off`, repeat the appropriate step (13, 14, 15, or 16) to turn on capture of that item.
14. After **diagport>**, type: `go`
    Press **Enter**.
15. Run the capture for one of the following periods of time:
    - If all microblock refresh times are one minute or less, run the capture for 5 minutes.
    - If any microblock refresh time is longer than 1 minute, run the capture for 5 times the longest microblock refresh time.
16. Type: `stop`
    Press **Enter**. Verify that you see diagport> before doing the next step.
17. After **diagport>**, type: `logout`
    Press **Enter**.

CARRIER CORPORATION ©2015
All rights reserved

18 To end the PuTTY session, click the X in the upper right corner.

19 In i-Vu®, clear the **Enable Telnet diagnostics** checkbox (see steps 3 and 4), then click **OK**.

20 Open the text file from the location you selected in step 11, and then verify that it legibly shows the same information that PuTTY displayed.

CARRIER CORPORATION ©2015
All rights reserved

Controller as a slave device on EIA-232/485

You can integrate an Carrier controller acting as a slave device into a third-party Modbus system.

Carrier		
Controllers	i-Vu® Link	i-Vu® Open Link
Driver	drv_ivulink_modbus_6-00-082* drv_ivuopenlink_std_6-00-082*	
Read/write capability	Can be read from and written to	
Third party		
Supported equipment	A master device that supports the Modbus (Modicon) protocol	
Network media type	EIA-232, EIA-485	
Quantity of devices you can physically connect to Port S2 on the Carrier controller	1 Modbus master	

*****You can download the latest drivers from the *Carrier Control Systems Support Site http://www.hvacpartners.com/*. Check the latest bulletin releases for new or updated drivers.

Before-you-begin checklist

You need the following items, information, and skills for the integration process:

- ☐ A list of the points in the Carrier controller that the Modbus device will read or write to.
- ☐ The Carrier controller's address on the Modbus network.
- ☐ The network media type required by the Modbus device: EIA-232 or EIA-485
- ☐ A port pinout/configuration of the Modbus device. Pinouts are usually available from the third-party manufacturer's representative or website.
- ☐ The Modbus network's baud rate, data bits, parity, and stop bits
- ☐ Verification that all communication properties have been set on the Modbus devices
- ☐ Verification of communications through the port that the Carrier controller will connect to
- ☐ Experience creating control programs in the Snap application
- ☐ Experience installing, wiring, setting up, and downloading to the Carrier controller

CARRIER CORPORATION ©2015
All rights reserved

The integration process

Follow the steps in this section to integrate a an i-Vu Link or i-Vu Open Link into a Modbus system. To install and network the Link, see the Installation and Start-up Guide.

1 Create a control program in the Snap application

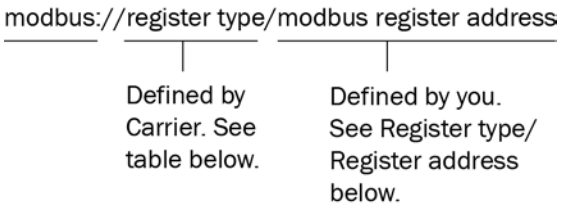
When you create your control program, use a Network I/O microblock for each value that you want to expose to the Modbus master.

Formatting a Modbus address

Use the information below to format a valid address in each microblock that you want to expose to the Modbus master.

CAUTION!
When integrating third-party devices into a i-Vu® system, most communication problems are caused by incorrect data or typing errors in the microblock's **Address** field.

Address format:



Example: modbus://UINT/40128

CARRIER CORPORATION ©2015
All rights reserved

Register type/Register address

When the Carrier controller is a slave device, you must expose its points so that they can be read or written to by a third-party master device.

For a third-party device to...	this kind of exposed value in the Carrier controller...	use this microblock...	with this register type...	and a Modbus register address in this range...
Write	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANI	Uint (not Unit)	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANI	Sint	**4**00001–**4**65535
	Value with decimal point Holding register (**4**XXXXX)	ANI	Float	**4**00001–**4**65535
	0 or 1 Coil Discrete (binary) output	BNI	Do	1–65535 [3]
Read	0 to 65,535 Unsigned 16-bit integer Input register (**3**XXXXX)	ANO	Uint (not Unit)	**3**00001–**3**65535
	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANO	Uint (not Unit)	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Input register (**3**XXXXX)	ANO	Sint	**3**00001–**3**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANO	Sint	**4**00001–**4**65535
	Value with decimal point Input register (**3**XXXXX)	ANO	Float	**3**00001–**3**65535
	Value with decimal point Holding register (**4**XXXXX)	ANO	Float	**4**00001–**4**65535
	0 or 1 Coil Discrete (binary) output	BNO	Do	1–65535 [3]
	0 or 1 Discrete (binary) input (**1**XXXXX)	BNO	Di	**1**00001–**1**65535

CARRIER CORPORATION ©2015
All rights reserved

For a third-party device to...	this kind of exposed value in the Carrier controller...	use this microblock...	with this register type...	and a Modbus register address in this range...

[1] If the register addresses in the points list are 5-digit numbers in the range **#**0001-**#**9999 (where **#** = the prefix **1**, **3**, or **4**), you may use the addresses as they are, or change them to fit the formats in the table above by shifting the prefix one digit to the left. For example, if you see the address **3**8129, you can also use **3**08129.

[2] Each Float uses 2 consecutive Modbus register addresses. Use the lower number in the microblock address.

[3] Enter the number without any leading zeros. For example, enter 125, not 000125.

See *Modbus Troubleshooting* (page 40) for more information on register addresses.

Editing a microblock address

You can edit a microblock address in the following places:

- In the Snap Property Editor
- In the i-Vu® interface on the microblock's **Properties** page > **Details** tab
- In the i-Vu® interface on the control program's **Properties** page > **Network Points** tab

2 Download the Modbus driver and control programs

If you do not already have the latest SAL library or drivers, download it from *Carrier Control Systems Support Site http://www.hvacpartners.com/* and save it to your computer.

To apply the SAL in the i-Vu interface:

1 Click **Main Menu** , then select **System Options** tree > **Update** tab.

NOTE Expand **Current Libraries (.sal)** to see the current SAL libraries and their revision. Compare them to what you downloaded and determine if any of them need updating.

2 Click Update Library and browse to the updated .sal file that you have saved on your computer, select the file, and click **Open**.

3 Click **Continue**.

4 When process is complete, the message appears **File added successfully**.

5 Click **Close**.

NOTE These changes are not applied until you have updated routers and controllers.

1 On the i-Vu® navigation tree, Right-click the router that you wish to update and click **Driver Properties**.

2 Select **Properties** page > **Update** tab.

3 If the database contains 2 or more routers, you must check **Change for all control programs of this type** in the **Controller** section.

CARRIER CORPORATION ©2015
All rights reserved

4 Click **Update**. A message appears **Changes the driver and screen file to use the current library version. Continue?**

NOTE If more than one router exists, the additional routers are listed below the **Update** button.

5 Click **OK.**

6 Click **Accept.**

7 Continue to update any other necessary devices.

3 Connect the Carrier controller to the third-party device

Wiring specifications

For...	Use...	Maximum Length
EIA-232	18–28 AWG; twisted pair preferable	50 feet (15.24 meters)
EIA-485, 2-wire	18–24 AWG twisted, shielded pair [1]	3000 feet (914.4 meters)

[1] Shielding provides noise immunity in an electrically noisy environment. Acceptable wiring types, in order of greatest to least noise immunity, are:
- 2-pair, each pair individually shielded
- 2-pair, single overall cable shield

To wire a third-party device

1 Turn off the i-Vu® Link/Open Link's power.

2 Check the communications wiring for shorts and grounds.

3 Wire the i-Vu® Link/Open Link's Port S2 to the third-party device, then set the S2 jumper. See table and notes below.

4 Turn on the i-Vu® Link/Open Link's power.

For...	Use i-Vu® Link/Open Link port...	Wire Carrier terminal...	...to third-party device terminal	Set the port's jumper(s) on i-Vu® Link/Open Link
EIA-232	S2	TX Rx Signal Ground	Rx Tx Gnd	EIA-232
EIA-485, 2-wire	S2	Net+ Net-	+ -	EIA-485 2-wire

CARRIER CORPORATION ©2015
All rights reserved

NOTES

- If you cannot determine the media type or connections of the third-party device, contact your third-party representative.
- Use the same polarity throughout the network segment.
- Repeaters are required for more than 31 devices. See your third-party device manufacturer's recommendations.
- To reduce communication and data errors, terminate each end of an EIA-485 network with a resistor whose value equals the network's characteristic impedance. Some third-party manufacturers provide a built-in resistor that you enable or disable with a jumper. Make sure that only devices at the end of a network have termination enabled.

  EXAMPLE If an EIA-485 2-wire network's characteristic impedance is 120 Ohms, terminate one pair by placing a 120 Ohm resistor across the **Net+** and **NET-** connectors of the i-Vu® Link/Open Link. Terminate the other pair by placing a 120 Ohm resistor across the **+** and **-** connectors of the furthest third-party controller.
- A solid receive light on the i-Vu® Link/Open Link indicates a wiring or polarity problem.

4 Set up the driver properties for Modbus EIA-232/485

1. On the i-Vu® navigation tree, right-click the i-Vu® Link or i-Vu® Open Link and select **Driver Properties**.
2. Expand **Protocols**, then select **Modbus**.
3. Under **Port Configuration**, check **Enable** under **Port S2**.
4. Set the remaining fields under **Port Configuration** to match the third-party device.
5. Under **Protocol Configuration**, set **Is this device a Master** to **No**.
6. In the **Modbus Slave Address for this port is** field, type the address of the Carrier controller on the third-party network.
7. Set the remaining fields using information from the third-party manufacturer's representative. Check **Details** for help.
8. Click **Apply**.
9. Select **Protocols** in the **Driver Properties** navigation tree.
10. In the **Protocol Status** table, verify that Modbus shows **Running** on Port S2

    If the status shows **Not Running** or the wrong port, verify that:

    - You set the controller's DIP switches correctly.
    - The control program has at least one valid Modbus address. The driver must detect at least one valid address to initiate Modbus communications.

CARRIER CORPORATION ©2015
All rights reserved

5 Verify the controller is set up correctly

1 On the i-Vu® navigation tree, select the controller.

2 Select the **Properties** page > **Network Points** tab.

If...	Then...
You see the point value you expect and the **Error** column shows **Initializing**	You have successfully exposed the Carrier controller's value to the third-party device.
All points show question marks instead of values	The i-Vu® application is not communicating with the Carrier controller or the control program. Troubleshoot the controller's communications. See the controller's *Installation and Start-up Guide*.
Some points show question marks instead of values	You may have exceeded the third-party points available in the controller. If so, do one of the following: • Use a controller that provides more third-party points. • Split the points between two control programs used in separate controllers. To determine the number of third-party points used in a controller: On the i-Vu® navigation tree, right-click on the controller and select **Driver Properties** > **Driver** > **Properties** > **Settings** tab, and then scroll to **Network Microblocks**. **Number of integration points requested** and **Number of integration points active** show how many third-party Network I/O microblocks the controller is using. These counts will differ if you exceed the controller's integration point limit. For example, if your i-Vu® Link/Open Link's control program includes 27 third-party points, your **Integration points requested** will be 27 and your **Integration points active** will be 25.
The point name is red	Select the controller on the i-Vu® navigation tree. Select **Properties** > **Network Points** tab. Look in the **Error** column for one of the following error codes and descriptions. • **1 - Communications Disabled for this Microblock** Enable the microblock's **Communications Enabled** field on the **Network Points** tab, on the microblock's **Properties** page > **Details** tab, or in the Snap application. • **3 - Address Error - Unknown Protocol Specified** Select the correct port on the Modbus driver page, verify that **Address** in the microblock is correct, or set the DIP switches correctly on the controller. • All other errors On the i-Vu® navigation tree, right-click the controller and select **Driver Properties**, Carrier controller's driver, then select **Protocols** > **Modbus Error Definitions**. Find the error and its possible solution.
A value is incorrect	Verify that the **Address** in the microblock is correct.

CARRIER CORPORATION ©2015
All rights reserved

If the above solutions do not resolve the problem, gather the following information for technical support:

- A diagnostic capture. See next topic.
- A screenshot of the **Driver Properties** > **Protocols** > **Properties** page and the **Driver** > **Protocols** > **Modbus** > **Properties** page
- A screenshot of the **Properties** page > **Network Points** tab showing addresses and errors
- Technical instructions for the third-party device, if available

To capture communication using PuTTY

Use PuTTY, a free open source terminal emulation program that works with all Windows operating systems, to capture communication between the controller and the Modbus device into a text file.

PREREQUISITES

- Download and install PuTTY from the *PuTTY website* (*http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html*).
- Get the Carrier controller's IP address.
- Get the longest microblock refresh time in the control program that has the error you are troubleshooting. In the i-Vu® interface, view the control program's **Properties** page > **Network Points** tab to see all the refresh times.

1. Connect your computer's Ethernet port to the controller's Ethernet port using one of the following:
   - A CAT5 or higher Ethernet crossover cable
   - A hub and a CAT5 or higher Ethernet straight-through cable
2. Ping the Carrier controller to verify communications between the controller and the computer you are using to get this capture.
3. On the i-Vu® navigation tree, right-click the controller, select **Driver Properties**, then select **Protocols**.
4. Click **Properties**, select **Enable Telnet diagnostics**, then click **Accept**.
5. Start PuTTY.
6. Enter the controller's IP address.
7. Select **Telnet** as the **Connection Type**.
8. Click the **Logging** option in the upper left corner of the left pane.
9. Select **Printable output**.
10. Select a location for the capture text file.
11. Click **Open** to start the session.
12. After **Login:>**, type: `diagport`
    Press **Enter**.
    - To capture data receipts, after **diagport>**, type: `modbus rx`

    Press **Enter**.
    - To capture data transmissions, after **diagport>**, type: `modbus tx`

    Press **Enter**.
    - To capture more detailed diagnostic messages, after **diagport>**, type: `modbus vmsg`

    Press **Enter**.
    - To capture error messages, after **diagport>**, type: `modbus emsg`

    Press **Enter**.

CARRIER CORPORATION ©2015
All rights reserved

13 Verify the displayed text shows:

```
modbus reporting level status:
      rx on
      tx on
      flush off
      fc1 off
      fc2 off
      vmsg on
      emsg on
          off
```

If `rx`, `tx`, `vmsg`, or `emsg` show `off`, repeat the appropriate step (13, 14, 15, or 16) to turn on capture of that item.

14 After **diagport>**, type: `go`
Press **Enter**.

15 Run the capture for one of the following periods of time:

- If all microblock refresh times are one minute or less, run the capture for 5 minutes.
- If any microblock refresh time is longer than 1 minute, run the capture for 5 times the longest microblock refresh time.

16 Type: `stop`
Press **Enter**. Verify that you see diagport> before doing the next step.

17 After **diagport>**, type: `logout`
Press **Enter**.

18 To end the PuTTY session, click the X in the upper right corner.

19 In i-Vu®, clear the **Enable Telnet diagnostics** checkbox (see steps 3 and 4), then click **OK**.

20 Open the text file from the location you selected in step 11, and then verify that it legibly shows the same information that PuTTY displayed.

CARRIER CORPORATION ©2015
All rights reserved

Controller as a client device on Ethernet

You can integrate Modbus devices into an i-Vu® system with a Carrier controller acting as an IP client in either of the following scenarios:

- The client controller reads and writes to points in a Modbus server.
- The client controller communicates with a Modbus gateway server to read and write to points in Modbus server devices.

Carrier		
Controllers	i-Vu® Link	i-Vu® Open Link
Driver	drv_ivulink_modbus_6-00-082* drv_ivuopenlink_std_6-00-082*	
Read/write capability	Can read from and write to the third-party equipment	
Third party		
Supported equipment	Any device that supports the Modbus (Modicon) protocol	
Network media type	Ethernet	
Quantity of devices you can connect to the Carrier client controller	Up to 30 Modbus TCP/IP connections.**	

***You can download the latest drivers from the *Carrier Control Systems Support Site http://www.hvacpartners.com/*. Check the latest bulletin releases for new or updated drivers.

**You may be able to connect more devices (256 maximum) if you consider the following:

- The lower the point count per device, the more devices you can connect.
- The more devices, the slower the communication between the Carrier controller and the Modbus devices.

A practical limit in most systems is approximately 100 devices. To connect more than 30 devices, change the **Close socket after each completed transaction** setting in the driver.

Before-you-begin checklist

You need the following items, information, and skills for the integration process:

- ☐ The IP addresses of the Modbus server
- ☐ The Unit ID of each Modbus device that contains points that the Carrier controller will read and write to, and a points list for each device that includes point register addresses and read/write capabilities. Points lists are usually available from the third-party manufacturer's representative or website.

CARRIER CORPORATION ©2015
All rights reserved

- ☐ Verification that all communication properties have been set on the Modbus devices
- ☐ Verification of Ethernet communication with each server that the Carrier controller will connect to
- ☐ Experience creating control programs in the Snap application
- ☐ Experience installing, wiring, setting up, and downloading to the Carrier controller

The integration process

Follow the steps in this section to integrate one or more third-party Modbus devices into an i-Vu® system using an i-Vu Link or i-Vu Open Link. To install and network a Link, see the Installation and Start-up Guide.

1 Create a control program in the Snap application

When you create your control program, use a Network I/O microblock for each third-party point.

TIP Verify the third-party manufacturer's register addressing pattern by establishing communication with a few points whose values you can physically manipulate before you spend time addressing the remaining integration points.

Formatting a Modbus address

Use the information below to format a valid address in each microblock that you use to read or write to a third-party point.

CAUTION!
When integrating third-party devices into a i-Vu® system, most communication problems are caused by incorrect data or typing errors in the microblock's **Address** field.

Address format:

mtcpip://register type/modbus register address/unit ID/IP address

Defined by Carrier. See Register type/ Register address below.	Provided in third-party points list. See Register type/ Register address below.	Set on the third-party device that contains the point. See Unit ID below.	IP address of the Modbus server

Example: mtcpip://UINT/40128/3/192.168.168.1

CARRIER CORPORATION ©2015
All rights reserved

Unit ID

If the register address is in...	the Unit ID is...
The Modbus server	0 or 255
A device on the Modbus server's serial network	1–254

Register type/Register address

To...	this kind of value...	use this microblock...	with this register type...	and a Modbus register address in this range... [1]
Read	0 to 65,535 Unsigned 16-bit integer Input register (**3**XXXXX)	ANI	Uint (not Unit)	**3**00001–**3**65535
	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANI	Uint (not Unit)	**4**00001–**4**65535
	0 to 4,294,967,296 Unsigned, 32-bit (long) integer Input register (**3**XXXXX)	ANI	Uint32	**3**00001–**3**65535
	0 to 4,294,967,296 Unsigned, 32-bit (long) integer Holding register (**4**XXXXX)	ANI	Uint32	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Input register (**3**XXXXX)	ANI	Sint	**3**00001–**3**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANI	Sint	**4**00001–**4**65535
	-2,147,483,648 to 2,147,483,647 Signed, 32-bit (long) integer Input register (**3**XXXXX)	ANI	Sint32	**3**00001–**3**65535
	-2,147,483,648 to 2,147,483,647 Signed, 32-bit (long) integer Holding register (**4**XXXXX)	ANI	Sint32	**4**00001–**4**65535
	Value with decimal point Input register (**3**XXXXX)	ANI	Float	**3**00001–**3**65535 [2]
	Value with decimal point Holding register (**4**XXXXX)	ANI	Float	**4**00001–**4**65535 [2]
	0 or 1 Coil / Discrete (binary) output	BNI (or ANI)	Do	1–65535 [3]
	0 or 1 Discrete (binary) input (**1**XXXXX)	BNI (or ANI)	Di	**1**00001–**1**65535

CARRIER CORPORATION ©2015
All rights reserved

To...	this kind of value...	use this microblock...	with this register type...	and a Modbus register address in this range... [1]
	0 or 1 Input register (**3**XXXXX)	BNI	BIT*n* (where *n* is a value 0-15 defined in points list)	**3**00001–**3**65535
	0 or 1 Holding register (**4**XXXXX)	BNI	BIT*n* (where *n* is a value 0-15 defined in points list)	**4**00001–**4**65535

To...	this kind of value...	use this microblock...	with this register type...	and a Modbus register address in this range... [1]
Write	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANO	Uint (not Unit)	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANO	Sint	**4**00001–**4**65535
	Value with decimal point Holding register (**4**XXXXX)	ANO	Float	**4**00001–**4**65535 [2]
	0 or 1 Coil Discrete (binary) output	BNO (or ANO)	Do	1–65535 [3]
	0 or 1 Holding register (**4**XXXXX)	BNO (or ANO)	BIT*n* (where *n* is a value 0-15 defined in points list)	**4**00001–**4**65535

[1] The Modbus register address (sometimes called register, address, code, or parameter) must be a decimal value. If the register addresses in the points list include the letters A-F, use a scientific calculator to convert these hexadecimal values to decimal values.

If the register addresses in the points list are 5-digit numbers in the range **#**0001-**#**9999 (where **#** = the prefix **1**, **3**, or **4**), you may use the addresses as they are, or change them to fit the formats in the table above by shifting the prefix one digit to the left. For example, if you see the address **3**8129, you can also use **3**08129.

[2] Each Float uses 2 consecutive Modbus register addresses. Use the lower number in the microblock address.

[3] Enter the number without any leading zeros. For example, enter 125, not 000125.

See *Modbus Troubleshooting* (page 40) for more information on register addresses.

Editing a microblock address

You can edit a microblock address in the following places:

- In the Snap Property Editor
- In the i-Vu® interface on the microblock's **Properties** page > **Details** tab
- In the i-Vu® interface on the control program's **Properties** page > **Network Points** tab

CARRIER CORPORATION ©2015
All rights reserved

2 Download the Modbus driver and control programs

If you do not already have the latest SAL library or drivers, download it from *Carrier Control Systems Support Site http://www.hvacpartners.com/* and save it to your computer.

To apply the SAL in the i-Vu interface:

1 Click **Main Menu** , then select **System Options** tree > **Update** tab.

NOTE Expand **Current Libraries (.sal)** to see the current SAL libraries and their revision. Compare them to what you downloaded and determine if any of them need updating.

2 Click Update Library and browse to the updated .sal file that you have saved on your computer, select the file, and click **Open**.

3 Click **Continue**.

4 When process is complete, the message appears **File added successfully**.

5 Click **Close**.

NOTE These changes are not applied until you have updated routers and controllers.

1 On the i-Vu® navigation tree, Right-click the router that you wish to update and click **Driver Properties**.

2 Select **Properties** page > **Update** tab.

3 If the database contains 2 or more routers, you must check **Change for all control programs of this type** in the **Controller** section.

4 Click **Update**. A message appears **Changes the driver and screen file to use the current library version. Continue?**

NOTE If more than one router exists, the additional routers are listed below the **Update** button.

5 Click **OK.**

6 Click **Accept.**

7 Continue to update any other necessary devices.

3 Connect the Carrier controller to the third-party device

Use one of the following CAT5 or higher Ethernet cables:

- A cross-over cable to connect the Carrier controller directly to the third-party Modbus device
- A straight-through cable to connect the controller to a hub or switch, and a second straight-through cable to connect the hub or switch to the third-party Modbus device

Maximum cable length: 328 feet (100 meters)

1 Turn off the Carrier controller's power.

2 Check the communications wiring for shorts and grounds.

3 Wire the Carrier controller's **Port E1** to the third-party device.

NOTE Port E1 will still be capable of BACnet communication.

4 Turn on the Carrier controller's power.

CARRIER CORPORATION ©2015
All rights reserved

4 Set up the driver properties for Modbus TCP/IP

1 On the i-Vu® navigation tree, right-click the i-Vu® Link or i-Vu® Open Link and select **Driver Properties**.

2 Expand **Protocols**, then select **Modbus TCP/IP**.

3 Under **Port Configuration**, select **TCP/IP** as the **Communication Type**.

4 The standard **Modbus TCP Port** is 502. If a different port is to be used, type the port number. Select the **Details** checkbox for help.

5 Under **Modbus Protocol Configuration**, set **This device is a Client** to **Yes**.

6 Set the remaining fields using information from the third-party manufacturer's representative. Check **Details** for help.

7 If the Carrier controller has more than 30 TCP/IP connections, under **Modbus TCP/IP Protocol Configuration**, change **Close socket after each completed transaction** to **Yes**.

NOTE If you connect more than 30 devices (256 maximum), consider the following:
- The lower the point count per device, the more devices you can connect.
- The more devices, the slower the communication between the Carrier controller and the Modbus devices.

A practical limit in most systems is approximately 100 devices.

8 Click **Accept**.

9 Select **Protocols** in the **Driver Properties** navigation tree.

10 In the **Protocol Status** table, verify that **mtcpip** shows **Running** on Port E1.

5 Verify the controller is set up correctly

1 On the i-Vu® navigation tree, select the controller.

2 Select the **Properties** page > **Network Points** tab.

If...	Then...
You see the point value you expect with no errors in the **Error** column	You have successfully established communication with the third-party device.
All points show question marks instead of values	The i-Vu® application is not communicating with the Carrier controller or the control program. Troubleshoot the controller's communications. See the controller's Installation and Start-up Guide.

CARRIER CORPORATION ©2015
All rights reserved

If...	Then...
Some points show question marks instead of values	You may have exceeded the third-party points available in the controller. If so, do one of the following: • Use a controller that provides more third-party points. • Split the points between two control programs used in separate controllers. To determine the number of third-party points used in a controller: On the i-Vu® navigation tree, right-click on the controller and select **Driver Properties** > **Driver** > **Properties** > **Settings** tab, and then scroll to **Network Microblocks**. **Number of integration points requested** and **Number of integration points active** show how many third-party Network I/O microblocks the controller is using. These counts will differ if you exceed the controller's integration point limit. For example, if your i-Vu® Link/Open Link's control program includes 27 third-party points, your **Integration points requested** will be 27 and your **Integration points active** will be 25.
The point name is red	Select the controller on the i-Vu® navigation tree. Select **Properties** > **Network Points** tab. Look in the **Error** column for one of the following error codes and descriptions. • **1 - Communications Disabled for this Microblock** Enable the microblock's **Communications Enabled** field on the **Network Points** tab, on the microblock's **Properties** page > **Details** tab, or in the Snap application. • **3 - Address Error - Unknown Protocol Specified** Select the correct port on the Modbus driver page, verify that **Address** in the microblock is correct, or set the DIP switches correctly on the controller. • All other errors On the i-Vu® navigation tree, right-click the controller and select **Driver Properties**, Carrier controller's driver, then select **Protocols** > **Modbus Error Definitions**. Find the error and its possible solution.
A value is incorrect	Verify that: • The **Address** in the microblock is correct. • The retrieved value is scaled properly, if necessary. For example, scaled from Celsius to Fahrenheit. Refer to the third-party manufacturer's documentation or the controller's Installation and Start-up Guide for scaling information.

If the above solutions do not resolve the problem, gather the following information for technical support:

- A diagnostic capture using Wireshark. See the following section for Wireshark instructions.

  NOTE If directed by Tech Support, you can use PuTTY to get a diagnostic capture. PuTTY instructions are after the Wireshark instructions.

- A screenshot of the driver's **Protocols** and **Modbus TCP/IP** pages.
- A screenshot of the **Properties** page > **Network Points** tab showing addresses and errors

CARRIER CORPORATION ©2015
All rights reserved

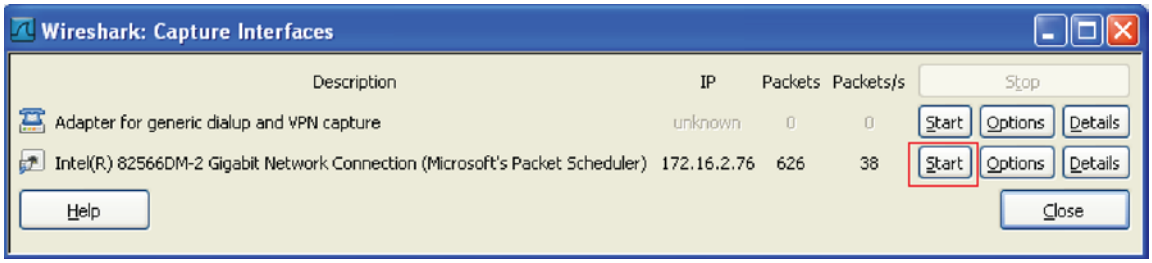
- All information from a Modstat copied into a text file.
  Right-click the Modstat, then select **Select All**. Press Ctrl+C to copy the information, then open Notepad and paste the information into a text file.
- Technical instructions for the third-party device, if available

To capture communication using Wireshark

Use Wireshark, a network analysis tool, to capture the Ethernet communication between the Carrier controller and the Modbus device.

PREREQUISITE Provide an Ethernet hub (not a switch) so that Wireshark can capture all Ethernet communication, not just broadcasts.

1. Download the latest version of Wireshark from the *Wireshark website* (*http://www.wireshark.org*).
2. Run the Wireshark install program, accepting all defaults. Include WinPcap in the installation.
3. Disconnect the network cable from the Carrier controller's Ethernet port, then plug the cable into the hub's **Uplink** port.
4. Use a separate Ethernet cable to connect the controller's Ethernet port to the hub.
5. Connect the Modbus device's Ethernet cable to the hub.
6. Connect the Ethernet port of the computer running Wireshark to the hub.
7. On the computer, click **Start** > **All Programs** > **Wireshark.**
8. From the menu bar, select **Capture** > **Interfaces**.



9. Click the **Start** button next to the adapter that is connected to the network. This starts the IP capture.

   TIP Choose the adapter that shows the **Packets** value changing.
10. Allow the capture to run long enough to ensure that there is sufficient data to allow a technician to review the problem.
11. On the menu bar, select **Capture** > **Stop** to stop the data capture.
12. Select **File** > **Save** and save the capture to a convenient location. Leave the **Save as type** default set to **Wireshark/tcpdump/... - libpcap (*.pcap, *.cap)**.
13. Send the file to Carrier Technical Support for analysis.

TIP You can color code the information in the Wireshark capture file based on user-defined criteria. See Wireshark's Help for instructions on setting up Coloring Rules.

CARRIER CORPORATION ©2015
All rights reserved

To capture communication using PuTTY

Use PuTTY, a free open source terminal emulation program that works with all Windows operating systems, to capture communication between the controller and the Modbus device into a text file.

PREREQUISITES

- Download and install PuTTY from the *PuTTY website* (*http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html*).
- Get the Carrier controller's IP address.
- Get the longest microblock refresh time in the control program that has the error you are troubleshooting. In the i-Vu® interface, view the control program's **Properties** page > **Network Points** tab to see all the refresh times.

1. Connect your computer's Ethernet port to the controller's Ethernet port using one of the following:
   - A CAT5 or higher Ethernet crossover cable
   - A hub and a CAT5 or higher Ethernet straight-through cable
2. Ping the Carrier controller to verify communications between the controller and the computer you are using to get this capture.
3. Right-click the controller, select **Driver Properties** and click **Protocols**.
4. Click **Properties**, check **Enable Telnet diagnostics**, then click **Accept**.
5. Start PuTTY.
6. Enter the controller's IP address.
7. Select **Telnet** as the **Connection Type**.
8. Click the **Logging** option in the upper left corner of the left pane.
9. Select **Printable output**.
10. Select a location for the capture text file.
11. Click **Open** to start the session.
12. After **Login:>**, type: `diagport`
    Press **Enter**.
13. To capture data receipts, after **diagport>**, type: `mtcpip rx`
    Press **Enter**.
14. To capture data transmissions, after **diagport>**, type: `mtcpip tx`
    Press **Enter**.
15. To capture more detailed diagnostic messages, after **diagport>**, type: `mtcpip vmsg`
    Press **Enter**.
16. To capture error messages, after **diagport>**, type: `mtcpip emsg`
    Press **Enter**.

CARRIER CORPORATION ©2015
All rights reserved

17 Verify the displayed text shows:

```
modbus reporting level status:
      rx on
      tx on
      flush off
      fc1 off
      fc2 off
      vmsg on
      emsg on
          off
```

If `rx`, `tx`, `vmsg`, or `emsg` show `off`, repeat the appropriate step (13, 14, 15, or 16) to turn on capture of that item.

18 After **diagport>**, type: `go`
Press **Enter**.

19 Run the capture for one of the following periods of time:

- If all microblock refresh times are one minute or less, run the capture for 5 minutes.
- If any microblock refresh time is longer than 1 minute, run the capture for 5 times the longest microblock refresh time.

20 Type: `stop`
Press **Enter**. Verify that you see diagport> before doing the next step.

21 After **diagport>**, type: `logout`
Press **Enter**.

22 To end the PuTTY session, click the X in the upper right corner.

23 In the i-Vu® interface, uncheck **Enable Telnet diagnostics** (see steps 3 and 4), then click **Accept**.

24 Open the text file from the location you selected in step 11, and then verify that it legibly shows the same information that PuTTY displayed.

CARRIER CORPORATION ©2015
All rights reserved

Controller as a server device on Ethernet

NOTE Modbus supports a single master on each network.

Carrier	
Controllers	i-Vu® Link/Open Link i-Vu® Open Link
Drivers	drv_ivulink_modbus_6-00-082* drv_ivuopenlink_std_6-00-082*
Read/write capability	Can be read from and written to
Third party	
Supported equipment	Any device that supports the Modbus (Modicon) protocol
Network media type	Ethernet
Quantity of devices you can connect to the Carrier controller	• Up to 30 Modbus TCP/IP connections.**

***You can download the latest drivers from the *Carrier Control Systems Support Site http://www.hvacpartners.com/*. Check the latest bulletin releases for new or updated drivers.

**You may be able to connect more devices (256 maximum) if you consider the following:

- The lower the point count per device, the more devices you can connect.
- The more devices, the slower the communication between the Carrier controller and the Modbus devices.
- A practical limit in most systems is approximately 100 devices. To connect more than 30 devices, change the **Close socket after each completed transaction** setting in the driver.

Before-you-begin checklist

You need the following items, information, and skills for the integration process:

- ☐ A list of the points in the Carrier controller that the Modbus client will read or write to.
- ☐ Verification that all communication properties have been set on the Modbus devices
- ☐ Verification of Ethernet communication with each Modbus client that the Carrier controller will connect to
- ☐ Experience creating control programs in the Snap application
- ☐ Experience installing, wiring, setting up, and downloading to the Carrier controller

CARRIER CORPORATION ©2015
All rights reserved

The integration process

Follow the steps in this section to integrate a an i-Vu Link or i-Vu Open Link into a Modbus system. To install and network the Link, see the Installation and Start-up Guide.

1 Create a control program in Snap

When you create your control program, use a Network I/O microblock for each value that you want to expose to the Modbus client.

Formatting a Modbus address

Use the information below to format a valid address for each value that you want to expose to the Modbus client.

CAUTION!
When integrating third-party devices into a i-Vu® system, most communication problems are caused by incorrect data or typing errors in the microblock's **Address** field.

Address format:

mtcpip://register type/modbus register address

	Defined by Carrier. See Register type/ Register address table below.	Provided in third-party points list. See Register type/ Register address below.

Example: mtcpip://UINT/40128

CARRIER CORPORATION ©2015
All rights reserved

Register type/Register address

When the Carrier controller is a server device, you must expose its points so that they can be read or written to by a third-party client device.

For a third-party device to...	this kind of exposed value in the Carrier controller...	use this microblock...	with this register type...	and a Modbus register address in this range...
Write	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANI	Uint (not Unit)	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANI	Sint	**4**00001–**4**65535
	Value with decimal point Holding register (**4**XXXXX)	ANI	Float	**4**00001–**4**65535
	0 or 1 Coil Discrete (binary) output	BNI	Do	1–65535 [3]
Read	0 to 65,535 Unsigned 16-bit integer Input register (**3**XXXXX)	ANO	Uint (not Unit)	**3**00001–**3**65535
	0 to 65,535 Unsigned 16-bit integer Holding register (**4**XXXXX)	ANO	Uint (not Unit)	**4**00001–**4**65535
	–32,768 to +32,767 Signed 16-bit integer Input register (**3**XXXXX)	ANO	Sint	**3**00001–**3**65535
	–32,768 to +32,767 Signed 16-bit integer Holding register (**4**XXXXX)	ANO	Sint	**4**00001–**4**65535
	Value with decimal point Input register (**3**XXXXX)	ANO	Float	**3**00001–**3**65535
	Value with decimal point Holding register (**4**XXXXX)	ANO	Float	**4**00001–**4**65535
	0 or 1 Coil Discrete (binary) output	BNO	Do	1–65535 [3]
	0 or 1 Discrete (binary) input (**1**XXXXX)	BNO	Di	**1**00001–**1**65535

CARRIER CORPORATION ©2015
All rights reserved

For a third-party device to...	this kind of exposed value in the Carrier controller...	use this microblock...	with this register type...	and a Modbus register address in this range...

[1] If the register addresses in the points list are 5-digit numbers in the range **#**0001-**#**9999 (where **#** = the prefix **1**, **3**, or **4**), you may use the addresses as they are, or change them to fit the formats in the table above by shifting the prefix one digit to the left. For example, if you see the address **3**8129, you can also use **3**08129.

[2] Each Float uses 2 consecutive Modbus register addresses. Use the lower number in the microblock address.

[3] Enter the number without any leading zeros. For example, enter 125, not 000125.

See *Modbus Troubleshooting* (page 40) for more information on register addresses.

Editing a microblock address

You can edit a microblock address in the following places:

- In the Snap Property Editor
- In the i-Vu® interface on the microblock's **Properties** page > **Details** tab
- In the i-Vu® interface on the control program's **Properties** page > **Network Points** tab

2 Download the Modbus driver and control programs

If you do not already have the latest SAL library or drivers, download it from *Carrier Control Systems Support Site http://www.hvacpartners.com/* and save it to your computer.

To apply the SAL in the i-Vu interface:

1. Click **Main Menu**, then select **System Options** tree > **Update** tab.

   NOTE Expand **Current Libraries (.sal)** to see the current SAL libraries and their revision. Compare them to what you downloaded and determine if any of them need updating.

2. Click Update Library and browse to the updated .sal file that you have saved on your computer, select the file, and click **Open**.
3. Click **Continue**.
4. When process is complete, the message appears **File added successfully**.
5. Click **Close**.

NOTE These changes are not applied until you have updated routers and controllers.

1. On the i-Vu® navigation tree, Right-click the router that you wish to update and click **Driver Properties**.
2. Select **Properties** page > **Update** tab.
3. If the database contains 2 or more routers, you must check **Change for all control programs of this type** in the **Controller** section.
4. Click **Update**. A message appears **Changes the driver and screen file to use the current library version. Continue?**

   NOTE If more than one router exists, the additional routers are listed below the **Update** button.
5. Click **OK.**

CARRIER CORPORATION ©2015
All rights reserved

6 Click **Accept.**

7 Continue to update any other necessary devices.

3 Connect the Carrier controller to the third-party device

Use one of the following CAT5 or higher Ethernet cables:

- A cross-over cable to connect the Carrier controller directly to the third-party Modbus device
- A straight-through cable to connect the controller to a hub or switch, and a second straight-through cable to connect the hub or switch to the third-party Modbus device

Maximum cable length: 328 feet (100 meters)

1 Turn off the Carrier controller's power.

2 Check the communications wiring for shorts and grounds.

3 Wire the Carrier controller's **Port E1** to the third-party device.

NOTE Port E1 will still be capable of BACnet communication.

4 Turn on the Carrier controller's power.

4 Set up the driver properties for Modbus TCP/IP

1 On the i-Vu® navigation tree, right-click the i-Vu® Link or i-Vu® Open Link and select **Driver Properties**.

2 Expand **Protocols**, then select **Modbus TCP/IP**.

3 Under **Port Configuration**, select **TCP/IP** as the **Communication Type**.

4 The standard **Modbus TCP Port** is 502. If a different port is to be used, type the port number. Select the **Details** checkbox for help.

5 Under **Modbus Protocol Configuration**, set **This device is a Client** to **No**.

6 Click **Apply**.

7 Select **Protocols**.

8 In the **Protocol Status** table, verify that **mtcpip** shows **Running** on Port E1.

CARRIER CORPORATION ©2015
All rights reserved

5 Verify the controller is set up correctly

1 On the i-Vu® navigation tree, select the controller.

2 Select the **Properties** page > **Network Points** tab.

If...	Then...
You see the point value you expect and the **Error** column shows **Initializing**	You have successfully exposed the Carrier controller's value to the third-party device.
All points show question marks instead of values	The i-Vu® application is not communicating with the Carrier controller or the control program. Troubleshoot the controller's communications. See the controller's Installation and Start-up Guide.
Some points show question marks instead of values	You may have exceeded the third-party points available in the controller. If so, do one of the following: • Use a controller that provides more third-party points. • Split the points between two control programs used in separate controllers. To determine the number of third-party points used in a controller: On the i-Vu® navigation tree, right-click on the controller and select **Driver Properties** > **Driver** > **Properties** > **Settings** tab, and then scroll to **Network Microblocks**. **Number of integration points requested** and **Number of integration points active** show how many third-party Network I/O microblocks the controller is using. These counts will differ if you exceed the controller's integration point limit. For example, if your i-Vu® Link/Open Link's control program includes 27 third-party points, your **Integration points requested** will be 27 and your **Integration points active** will be 25.
The point name is red	Select the controller on the i-Vu® navigation tree. Select **Properties** > **Network Points** tab. Look in the **Error** column for one of the following error codes and descriptions. • **1 - Communications Disabled for this Microblock** Enable the microblock's **Communications Enabled** field on the **Network Points** tab, on the microblock's **Properties** page > **Details** tab, or in the Snap application. • **3 - Address Error - Unknown Protocol Specified** Select the correct port on the Modbus driver page, verify that **Address** in the microblock is correct, or set the DIP switches correctly on the controller. • All other errors On the i-Vu® navigation tree, right-click the controller and select **Driver Properties**, Carrier controller's driver, then select **Protocols** > **Modbus Error Definitions**. Find the error and its possible solution.
A value is incorrect	Verify that the **Address** in the microblock is correct.

CARRIER CORPORATION ©2015
All rights reserved

If the above solutions do not resolve the problem, gather the following information for technical support:

- A diagnostic capture using Wireshark. See the following section for Wireshark instructions.

  NOTE If directed by Tech Support, you can use PuTTY to get a diagnostic capture. PuTTY instructions are after the Wireshark instructions.

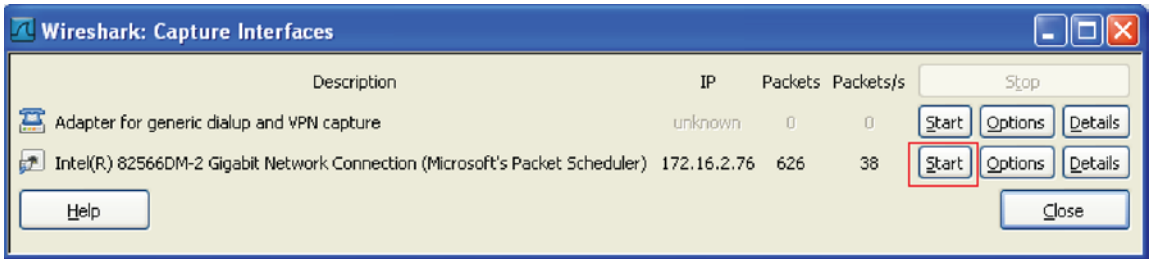
- A screenshot of the driver's **Protocols** and **Modbus TCP/IP** pages.
- A screenshot of the **Properties** page > **Network Points** tab showing addresses and errors
- All information from a Modstat copied into a text file.
  Right-click the Modstat, then select **Select All**. Press Ctrl+C to copy the information, then open Notepad and paste the information into a text file.
- Technical instructions for the third-party device, if available

To capture communication using Wireshark

Use Wireshark, a network analysis tool, to capture the Ethernet communication between the Carrier controller and the Modbus device.

PREREQUISITE Provide an Ethernet hub (not a switch) so that Wireshark can capture all Ethernet communication, not just broadcasts.

1. Download the latest version of Wireshark from the *Wireshark website* (*http://www.wireshark.org*).
2. Run the Wireshark install program, accepting all defaults. Include WinPcap in the installation.
3. Disconnect the network cable from the Carrier controller's Ethernet port, then plug the cable into the hub's **Uplink** port.
4. Use a separate Ethernet cable to connect the controller's Ethernet port to the hub.
5. Connect the Modbus device's Ethernet cable to the hub.
6. Connect the Ethernet port of the computer running Wireshark to the hub.
7. On the computer, click **Start** > **All Programs** > **Wireshark.**
8. From the menu bar, select **Capture** > **Interfaces**.



9. Click the **Start** button next to the adapter that is connected to the network. This starts the IP capture.

   TIP Choose the adapter that shows the **Packets** value changing.

10. Allow the capture to run long enough to ensure that there is sufficient data to allow a technician to review the problem.

CARRIER CORPORATION ©2015
All rights reserved

11 On the menu bar, select **Capture** > **Stop** to stop the data capture.

12 Select **File** > **Save** and save the capture to a convenient location. Leave the **Save as type** default set to **Wireshark/tcpdump/... - libpcap (*.pcap, *.cap)**.

13 Send the file to Carrier Technical Support for analysis.

TIP You can color code the information in the Wireshark capture file based on user-defined criteria. See Wireshark's Help for instructions on setting up Coloring Rules.

To capture communication using PuTTY

Use PuTTY, a free open source terminal emulation program that works with all Windows operating systems, to capture communication between the controller and the Modbus device into a text file.

PREREQUISITES

- Download and install PuTTY from the *PuTTY website* (*http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html*).
- Get the Carrier controller's IP address.
- Get the longest microblock refresh time in the control program that has the error you are troubleshooting. In the i-Vu® interface, view the control program's **Properties** page > **Network Points** tab to see all the refresh times.

1 Connect your computer's Ethernet port to the controller's Ethernet port using one of the following:
   - A CAT5 or higher Ethernet crossover cable
   - A hub and a CAT5 or higher Ethernet straight-through cable

2 Ping the Carrier controller to verify communications between the controller and the computer you are using to get this capture.

3 Right-click the controller, select **Driver Properties** and click **Protocols**.

4 Click **Properties**, check **Enable Telnet diagnostics**, then click **Accept**.

5 Start PuTTY.

6 Enter the controller's IP address.

7 Select **Telnet** as the **Connection Type**.

8 Click the **Logging** option in the upper left corner of the left pane.

9 Select **Printable output**.

10 Select a location for the capture text file.

11 Click **Open** to start the session.

12 After **Login:>**, type: `diagport`
Press **Enter**.

13 To capture data receipts, after **diagport>**, type: `mtcpip rx`
Press **Enter**.

14 To capture data transmissions, after **diagport>**, type: `mtcpip tx`
Press **Enter**.

15 To capture more detailed diagnostic messages, after **diagport>**, type: `mtcpip vmsg`
Press **Enter**.

CARRIER CORPORATION ©2015
All rights reserved

16 To capture error messages, after **diagport>**, type: `mtcpip emsg`
Press **Enter**.

17 Verify the displayed text shows:

```
modbus reporting level status:
      rx on
      tx on
      flush off
      fc1 off
      fc2 off
      vmsg on
      emsg on
          off
```

If `rx`, `tx`, `vmsg`, or `emsg` show `off`, repeat the appropriate step (13, 14, 15, or 16) to turn on capture of that item.

18 After **diagport>**, type: `go`
Press **Enter**.

19 Run the capture for one of the following periods of time:

- If all microblock refresh times are one minute or less, run the capture for 5 minutes.
- If any microblock refresh time is longer than 1 minute, run the capture for 5 times the longest microblock refresh time.

20 Type: `stop`
Press **Enter**. Verify that you see diagport> before doing the next step.

21 After **diagport>**, type: `logout`
Press **Enter**.

22 To end the PuTTY session, click the X in the upper right corner.

23 In the i-Vu® interface, uncheck **Enable Telnet diagnostics** (see steps 3 and 4), then click **Accept**.

24 Open the text file from the location you selected in step 11, and then verify that it legibly shows the same information that PuTTY displayed.

CARRIER CORPORATION ©2015
All rights reserved

Modbus Troubleshooting

The following troubleshooting information depends on whether your Carrier controller is a master/client or a slave/server.

In most cases, a diagnostic capture can help determine the problem. See *To capture communication using …* in the appropriate section of this document.

If the Carrier controller is a Serial Master or Ethernet Client

Register addresses

Because Modbus register address formats differ from one manufacturer to another, the most common problems in a Modbus integration are caused by a microblock address having an incorrect register address.

If the Carrier controller is a serial master or Ethernet client, the register address that you include in a microblock address is the register address that you want to read from/write to in the slave/server. Microblocks support Modbus register addresses that consist of a base register address (1–65535) and a one digit prefix (except coils) that defines the type of value.

This type of value…	Has the prefix…	Example of register address
Discrete input	1	**1**03459 or **1**3459
Input register	3	**3**04564 or **3**4564
Holding register	4	**4**01563 or **4**1563
Coil	No prefix	

Base address
Most manufacturers begin counting register addresses at 1; some begin at 0. Other than contacting the manufacturer to get clarification, you can look at the register address list to see if the addressing in any table starts at 0. If any table's addressing begins at 0, you need to add 1 to each register address in order to read the right register. For example, if you are given a register address of **4**0100, you must enter the address as **4**0101 to read the correct register.

Prefix
Some manufacturers include the prefix in their register address. If you add another prefix, you will have an invalid register address. For example, if the third-party manufacturer shows a holding register address is **4**0123 and you add the prefix **4**, your register address will be **4**40123, which is an invalid address.

CARRIER CORPORATION ©2015
All rights reserved

Errors

If the **Properties** page > **Network Points** tab shows errors, look for the error in the table below. To see a more extensive list, go to the **Driver Properties** tree > **Protocols** > **Modbus Error Definitions**.

Error code/message	Possible solution(s)
Error ## - Modbus Exception Code - XXXX	Where: **##** = a number from 51 to 61, and **XXXX** = the exception code description A server/slave device returns a Modbus Exception Code when it cannot respond to the master/client. The most common are: • **ILLEGAL FUNCTION**: The master/client sends an unsupported function code to the slave/server. For example, the master/client tries to query with function code 16, which is not supported by the slave/server. **NOTE** Some manufacturers do not support: Function Code 15—Write Multiple Coils, or Function Code 16—Write Multiple Registers To resolve this, go to the **Driver Properties** tree > **Protocols** > **Modbus TCP/IP**. Set **Restrict coil and register commands** to **Yes**. • **ILLEGAL DATA ADDRESS**: The master/client sends an invalid register address to the slave/server. For example, the master/client tries to read register 500, but the highest register address is 100. To resolve this error, correct the register address. • **ILLEGAL DATA VALUE**: The master/client sends an invalid value to the slave/server. For example, the master/client tries to write a decimal value 150 to a Modbus Coil. To resolve this error, change the value you are writing to a value the register can accept.
Error 94 - Response Err – Invalid Exception Code	Although rare, the Exception Code returned by the Modbus slave/server device may be incorrect. Valid Exception Code range is 1 to 11 (except 9). Use a diagnostic capture to determine the invalid exception code. Then contact the third-party manufacturer for a resolution.
Error 72 – Addr Err – Invalid Register Number Prefix or **Error 73 – Addr Err – Invalid Register Number Range**	See *Register Addresses* (page 40).
Error 78 – Addr Err – Invalid Duplicates Found	You cannot have multiple ANO/BNO microblocks in the same controller write to the same address. If you have duplicates, only one will be processed. Remove all duplicates, and then download memory. **NOTE** You can have multiple ANI/BNI microblocks read from the same address.

CARRIER CORPORATION ©2015
All rights reserved

Error code/message	Possible solution(s)
Error 79 – Addr Err – Possible Overlapped Register	This error indicates registers that share a common address. For example: Modbus://FLOAT/401000/58 Modbus://UINT/401001/58 In this example, the FLOAT register type uses two register addresses, 1000 and 1001. So, the UINT's register address overlaps the FLOAT's 1001 address. The last microblock processed will get an error. To find the issue, the application programmer should check the previous and next sequential microblock register addresses.
Error 11 – Unable to allocate packet	Although rare, you may see this error if you have a large number of network points (more than 250) and most microblocks have a **Refresh Time** of 10 seconds or less. To resolve this, increase the **Refresh Time** on non-critical points. Critical points can still be updated more frequently. As described below, you can use diagnostic captures to fine-tune your system to make the most of the available bandwidth. See *To capture communication using …* in the appropriate section of this document. Only turn on the vmsg (verbose message) to see how the points are queried. Serial Master By looking at the capture, the engineer can follow how efficient the query may be. Look at the **bolded** packets below. The Master reads from device 58, Function Code 3 (Read Holding Reg) and register address 1 through 3, for the total of 3 registers. This packet is repeated twice in sequence, which may not be desirable. To resolve this, change the refresh time of microblocks with register address 2 and 3 to 10 minutes (or larger). Register addresses 2 and 3 will be read and updated when register address 1 is read and will only trigger an update if register address 1 is removed from the application. Observe the capture and make sure that the repetition is resolved. MODBUS<VM>READ Dev:58 - FC:3 - Reg:601 -> 603 - Qty:3 **MODBUS<VM>READ Dev:58 - FC:3 - Reg:1 -> 3 - Qty:3** **MODBUS<VM>READ Dev:58 - FC:3 - Reg:1 -> 3 - Qty:3** MODBUS<VM>READ Dev:58 - FC:3 - Reg:601 -> 603 - Qty:3 MODBUS<VM>READ Dev:58 - FC:3 - Reg:1 -> 3 - Qty:3 MODBUS<VM>READ Dev:58 - FC:3 - Reg:401 -> 406 - Qty:6 MODBUS<VM>READ Dev:58 - FC:4 - Reg:2001 -> 2009 - Qty:9 MODBUS<VM>READ Dev:58 - FC:4 - Reg:1001 -> 1009 - Qty:9 MODBUS<VM>READ Dev:58 - FC:4 - Reg:601 -> 603 - Qty:3 MODBUS<VM>READ Dev:58 - FC:4 - Reg:301 -> 306 - Qty:6 MODBUS<VM>READ Dev:58 - FC:4 - Reg:201 -> 206 - Qty:6 MODBUS<VM>READ Dev:58 - FC:4 - Reg:101 -> 103 - Qty:3 MODBUS<VM>READ Dev:58 - FC:4 - Reg:1 -> 3 - Qty:3

CARRIER CORPORATION ©2015
All rights reserved

Error code/message	Possible solution(s)
	IP Client Use the same techniques described above for the Serial Master. Look at the **bolded** packets below. The client reads from Device IP 161.145.174.36 with device address 0. It reads register 65532 through 65535 twice sequentially. To resolve this, change the refresh time to 10 minutes (or larger) for each microblock in the group except for the first microblock with register address 65532. Register addresses 65533 through 65535 will be read and updated when register address 65532 is read, and will only trigger an update if register address 65532 is removed from the application. MTCPIP<VM>Rx response ID:44 MTCPIP<VM>Rx response ID:45 MTCPIP<VM>READ Dev IP:161.145.174.236 - Addr:0 - FC:3 - Reg:40009 -> 40016 - Qty:8 - ID:55 MTCPIP<VM>READ Dev IP:161.145.174.236 - Addr:0 - FC:3 - Reg:40000 -> 40007 - Qty:8 - ID:56 MTCPIP<VM>READ Dev IP:161.145.174.236 - Addr:0 - FC:3 - Reg:1 -> 12 - Qty:12 - ID:57 MTCPIP<VM>Rx response ID:46 MTCPIP<VM>Rx response ID:47 MTCPIP<VM>READ Dev IP:161.145.174.236 - Addr:0 - FC:2 - Reg:65532 -> 65535 - Qty:4 - ID:58 **MTCPIP<VM>READ Dev IP:161.145.174.236 - Addr:0 - FC:1 - Reg:65532 -> 65535 - Qty:4 - ID:59** **MTCPIP<VM>READ Dev IP:161.145.174.236 - Addr:0 - FC:1 - Reg:65532 -> 65535 - Qty:4 - ID:60** MTCPIP<VM>Rx response ID:48 MTCPIP<VM>Rx response ID:49
Error 86 – Unable to update point value	The microblock value is not updating. Serial Master In rare cases, the master may receive a valid response but cannot match it to any microblock. The only way to determine this is to review a diagnostic capture. To resolve, in the i-Vu® interface, go to the controller's **Driver Properties** tree > **Protocols** > **Modbus** page, and increase the **Interpacket Delay**. This solution is not desirable if the controller's point count is high. IP Client Because Modbus TC/IP does not have a timeout defined, this error may occur when the client communicates with a very slow server. If a client microblock with a low refresh time (1 second) sends a query every second but the server requires more than 2 seconds to respond, this error will be set on the microblocks after a few cycles of queries. Get a diagnostic capture using Wireshark to see the timing between query and response. Increase the refresh time to allow the server to respond before a new request is sent. This solution will slow communication between the server and client controllers.

CARRIER CORPORATION ©2015
All rights reserved

Error code/message	Possible solution(s)
Error 40 – No response	Serial Master If the error is persistent, check the wiring, communication baud rate, and device address. If it is intermittent, in the i-Vu® interface, go to the controller's **Driver Properties** tree > **Protocols** > **Modbus** page, and try changing the following settings: • Increase the **Interpacket Delay** in increments of 10 ms. Maximum recommended is 100 ms, depending on the size of control program. The bigger the control program is, the smaller this setting should be. • If the query reads multiple registers, try reducing the size of **Maximum Registers to Request**. Use the diagnostics capture to determine where the issue is most likely to occur. If you need to reduce multiple registers, set the **Maximum Registers to Request** and/or **Maximum Coils/DIs to Request** to 1. You can optimize later if needed. • If the query writes to multiple registers, you may want to inhibit multiple write altogether. Use the diagnostics capture to determine where the issue is most likely to occur. In most cases, the slave device is more susceptible to multiple writes. To inhibit multiple registers writes, set **Restrict coil and register commands** to **Yes**. IP Client TCP/IP communication is successful, but the Modbus packet never gets a response from the Modbus server. Talk to the third-party vendor to determine the reason the server is not responding.
No microblock error, but a float value is not correct	FLOAT register types use 2 register addresses. The order in which they are processed differs from one manufacturer to another. If you see incorrect float values, go to the **NET** tree > the controller > **Driver** > **Protocols** > **Modbus TCP/IP**. Change the value of the **Reverse Float/32 Bit Integer** field. **CAUTION** This setting is applied globally for all float values in the controller. If you have some float registers that are to be processed one way and others that are to be processed the other way, you must use separate master/client devices to talk to them.

CARRIER CORPORATION ©2015
All rights reserved

If the Carrier controller is a Serial Slave or Ethernet Server

Register addresses

If the Carrier controller is a slave/server, register addresses are 1-based, not 0-based, and do not require any adjustment as long as the serial master or Ethernet client adheres to the Modbus protocol implementation described on the Modbus website (www.modbus.org).

If the Carrier controller is a slave/server, the register address that you include in a microblock address is the register address that you want to expose to master/client. Microblocks support Modbus register addresses that consist of a base register address (1–65535) and a one digit prefix (except coils) that defines the type of value.

This type of value...	Has the prefix...	Example of register address
Discrete input	1	**1**03459 or **1**3459
Input register	3	**3**04564 or **3**4564
Holding register	4	**4**01563 or **4**1563
Coil	No prefix	

Prefix
Some manufacturers include the prefix in their register address. If you add another prefix, you will have an invalid register address. For example, if the third-party manufacturer shows a holding register address is **4**0123 and you add the prefix **4**, your register address will be **4**40123, which is an invalid address.

Errors

If the **Properties** page > **Network Points** tab shows errors, look for the error in the table below. To see a more extensive list, go to the controller's **Driver Properties** tree > **Protocols** > **Modbus Error Definitions**.

Error code/message	Possible solution(s)
Error 72 – Addr Err – Invalid Register Number Prefix or **Error 73 – Addr Err – Invalid Register Number Range**	See *Register Addresses* (page 45).
Error 78 – Addr Err – Invalid Duplicates Found	Ensure all addresses are unique.
No microblock error, but the value displayed at the master/client is not correct	FLOAT register types use 2 register addresses. The order in which they are processed differs from one manufacturer to another. If you see incorrect float values, go to the controller's **Driver Properties** tree > **Protocols** > **Modbus TCP/IP**. Change the value of the **Reverse Float/32 Bit Integer** field.

CARRIER CORPORATION ©2015
All rights reserved

Appendix A - Modbus Protocol Conformance Statement

The following Modbus features and commands are supported by the Carrier Modbus drivers. See the *Modbus website* (*http://www.modbus.org*) for complete Modbus protocol information.

Serial connection:

Modes	RTU[1] ASCII	
Media type	EIA-232 EIA-485, 2-wire[1]	
Baud rate	1200 2400 4800	9600[1] 19200[1] 38400[1]
Data bits	7 8[1]	
Parity	None[1] Odd Even	
Stop bits	1[1] 2	

[1] Most commonly used value(s)

Ethernet connection:

Communication Type	TCP/IP
Media Type	Ethernet
Baud rate	10 Mbps/100 Mbps
Maximum TCP/IP Connections	30 [2]

[2] You may be able to connect more devices (256 maximum) if you consider the following:
- The lower the point count per device, the more devices you can connect.
- The more devices, the slower the communication between the Carrier controller and the Modbus devices.

A practical limit is approximately 100 devices. To connect more than 30 devices, change the **Close socket after each completed transaction** setting in the driver.

Serial or Ethernet:

Function codes - command	Register address range
01 - Read coils	1-65535
02 - Read discrete inputs	**1**00001-**1**65535

CARRIER CORPORATION ©2015
All rights reserved

Function codes - command	Register address range
03 - Read holding registers	**4**00001-**4**65535
04 - Read input registers	**3**00001-**3**65535
05 - Write single coil	
06 - Write single register	
15 - Write multiple coils	
16 - Write multiple registers	

CARRIER CORPORATION ©2015
All rights reserved

Document revision history

Important changes to this document are listed below. Minor changes such as typographical or formatting errors are not listed.

Date	Topic	Change description	Code*
7/7/15	Latest driver location referenced throughout the document	Corrected website path to current drivers.	C-TS-E-RR
	Formatting a Modbus address (Controller as a server device on Ethernet)	Corrected address format.	C-TS-E-RR
	Controller as a server device on Ethernet	Corrected quantity of devices you can connect to	C-TS-E-RR
	Set up Driver properties for EIA-232/485 (for Ehernet)	Title corrected to Set up Driver properties for TCP/IP	C-TS-E-RR
8/5/14	Formatting a Modbus address	Added footnote [3] explaining not to use leading zeros on the Coil/Discrete register address	A-AE-HP-E-HP

* For internal use only

CARRIER CORPORATION ©2015
All rights reserved

CARRIER CORPORATION ©2015
A member of the United Technologies Corporation family · Stock symbol UTX · Catalog No. 11-808-515-01 · 7/7/2015