BACnet/Modbus Carrier Translator

Overview and Configuration Manual

808-356 01/05

Introduction

Introduction

About this Manual

This manual contains information about the functions of the Carrier Translator module with BACnet/Modbus communication and how the user configures the BACnet/Modbus Carrier Translator to perform those functions.

The manual is divided into the following sections:

Introduction
Operating Characteristics
Configuration
Maintenance
Appendix A and B

The Introduction consists of this description of the manual.

The Operating Characteristics section contains a description of the Carrier Translator hardware and a summary description of its configuration and maintenance tables.

The Configuration section contains detailed lists of the decisions for each Carrier Translator configuration table. Each list entry includes the decision's purpose, the range of values that may be used, and the default values that will appear in the decision if it is not configured by the user.

The Maintenance section contains detailed lists of the decisions for each Carrier Translator maintenance table. Each list entry includes the decision's purpose and the range of values that may be displayed.

The Appendixes contain BACnet and Modbus compliance statements as well as tables that indicate BACnet and Modbus point mapping. Configuration sheets are provided so that they may be photocopied for use as worksheets and hard copy records when configuring the Carrier Translator.

Operating Characteristics

Operating Characteristics

The BACnet/MODBUS Carrier Translator is a micro controller-based module that provides the ability to integrate Carrier CCN-based controllers into BACnet or Modbus-based networks.

The BACnet/MODBUS Carrier Translator (33CNTRAN485) provides CCN-to-Modbus Remote Terminal Unit (RTU) or CCN-to-BACnet Master-Slave/Token-Passing (MS/TP) protocol conversion over BACnet /MODBUS.

The Carrier Translator can be mounted in the controls section of any CCN equipment and converts the CCN-based controller data to BACnet or Modbus. The Carrier Translator is outdoor duty rated and contains a CCN BACnet/MODBUS connector.

When connected to a CCN controller, the BACnet/MODBUS Carrier Translator allows a third party BACnet or Modbus device to read and write to the CCN controller's mapped status display, time schedule, and setpoint schedule data. Note that status display write access is subject to the CCN equipment controller's defined read/write access for each status display item.

When configured for BACnet, all points will be mapped as either AV or BV BACnet objects. The only property that can be written to is the Present Value.

If interfacing to a CCN device that contains multiple identical setpoint tables (for example, the Comfort Controller), the Translator will only expose the first instance of a setpoint point name.

Default Address and Baud Rate

The BACnet/MODBUS Carrier Translator's default CCN address is 0,200 (bus number, system element number). The default CCN baud rate is 9600 bps.

Note: To modify the BACnet/MODBUS Carrier Translator's BACnet/Modbus address, a Carrier user interface (such as ComfortVIEW or the Network Service Tool) must be used. The device has a default address of 16 on the BACnet or Modbus side.

LED Indicators

The Carrier Translator has three LEDs that are used to indicate operational status:

LED	Color	Indicates
Status	Red	Operating, initialization and configuration status. The LED blinks at a 2 Hz rate when initializing and at 1 Hz when operating correctly.
CCN	Yellow	The Carrier Translator is sending CCN communication messages to the connected CCN controller. If the connected CCN controller is responding, its CCN LED will blink when a message is sent back to the Carrier Translator.
BACnet/ Modbus	Green	The Carrier Translator is sending BACnet/Modbus communication messages to the third party network.

Carrier Translator Configuration Tables

The Carrier Translator contains the configuration tables listed below. For descriptions of the decisions in each table, refer to the Configuration section of this manual. The purpose of each table is summarized on the following pages.

Carrier Translator Configuration Table (CONFIG)
Point Mapping Table 1 (PT_CFGl)
Point Mapping Table 2 (PT_CFG2)
Setpoint Mapping Table (SP_CFG)
Occupancy Mapping Table (TS_CFG)

Carrier Translator Configuration Table

The Carrier Translator Configuration Table (CONFIG) contains decisions used to specify the following:

BACnet/MODBUS Protocol (BACnet or Modbus)
Display Units (US or Metric)
CCN Device Address
Auto-mapping
Reset Points Profile

Point Mapping Tables

The Point Mapping Tables (PT_CFGl and PT_CFG2) are used to map or associate CCN points with BACnet objects or Modbus registers. The Setpoint Mapping Table (SP_CFG1) maps the individual setpoint names, while the Occupancy Mapping Table (TS_CFG) maps the occupancy table names.

Device Configuration Table

The Carrier Translator contains a Device Configuration Table (CtlrID). By changing the information that appears in this table, you can change the name, description, and location that appears for the Carrier Translator in the CCN front end's (example: ComfortWORKS/ComfortVIEW) Controller List.

Carrier Translator Maintenance Tables

The Carrier Translator contains the following maintenance tables:

Communication Status Table (COMMSTAT)
Messages Table (MESSAGES)
Point Tables (PT_MNTl-6)
Occupancy Table (TS_MNT)
Setpoint Table (SP_MNT)

Communication Status Maintenance Table

The maintenance values displayed in this table are read-only values that show diagnostic data about BACnet or Modbus-to-CCN system element communication.

Messages Maintenance Table

The maintenance values displayed in this table are read-only values that show diagnostic data about BACnet/Modbus-to-CCN system element communication messages.

Point Maintenance Tables

The maintenance values displayed in these tables are read-only values that show the current value of the CCN points that have been mapped to the BACnet variables or Modbus registers as the points exist in the CCN system element. These values are updated every 30 seconds.

Occupancy Maintenance Table

The maintenance values in this table show the current value of CCN occupancy table points.

Setpoint Maintenance Table

The values in this table show the current value of CCN setpoint table points.

Configuration

Configuration

The BACnet/Modbus Carrier Translator's operation is controlled by decisions entered in a group of configuration tables. The Carrier Translator contains the following configuration tables:

CTLR_ID	Controller Identification Table
CONFIG	Carrier Translator Configuration Table
PT_CFG1	Point Mapping Configuration Table 1
PT_CFG2	Point Mapping Configuration Table 2
SP_CFG	Setpoint Mapping Configuration Table
TS_CFG	Occupancy Mapping Configuration Table

Configuration Process

BACnet/Modbus Carrier Translator configuration and start-up consists of four main processes:

1. Physically connect the BACnet/Modbus Carrier Translator to the CCN system element.
2. Select the appropriate BACnet/Modbus protocol and protocol address and specify the address of the PIC. You do this in the CONFIG Table.
3. Map, or associate, CCN points and setpoint and time schedules with BACnet/Modbus points by configuring the PT_CFG, SP_CFG, and TS_CFG Tables if necessary, or simply look at these tables and accept the automatic point mapping.
4. Commission the Carrier Translator on the BACnet/Modbus network.

Each of the above-listed processes is discussed below.

1. Connect the target CCN system element to the Carrier Translator's CCN connector. Confirm that the element is powered and communicating on the CCN Bus.
2. Using a CCN front end (such as a Network Service Tool or ComfortVIEW) access the Carrier Translator’s CONFIG Table and select the appropriate BACnet/Modbus protocol by entering a 0 or a 1 into the CONFIG Table's Protocol decision. You should also enter all items relating to the Protocol Address and communication configuration.

Note: All values relating to the BACnet/Modbus configuration should be coordinated with the systems integrator for each site.

3. Enter appropriate values into all CONFIG Table decisions and download the CONFIG Table into the Carrier Translator.

   If the Carrier Translator is on a CCN bus with elements other than the single CCN device with which it is communicating, you must:

   - Set Bus and Element #s
   - Set the Reset Points Profile decision to *Yes*
   - Download.

   This chapter's Carrier Translator Configuration (CONFIG) Table section contains explanations of and allowable entries and default values for each CONFIG Table decision.

   After approximately 5 seconds, the Carrier Translator should identify the target CCN system element and begin to download its tables. This takes 10 to 40 seconds, depending on the number of tables in the target CCN system element. During this period, the Yellow LED will flash to indicate CCN bus activity. When the flashing stops, the download is complete.

   At this point the Carrier Translator will automatically populate the PT_CFG1, PT_CFG2, SP_CFG, and TS_CFG Tables.

   Note: Do not cycle power to the Carrier Translator within 30 seconds of downloading the CONFIG Table or any of the mapping tables. Be sure that the Yellow CCN LED is NOT flashing.

Map CCN Points with RS-485 Variables

1. Using a CCN front end, access the following point configuration tables:

   PT_CFG1
   PT_CFG2

2. You will now have the opportunity to manually replace any of the automatically-selected CCN point names.

   Note: Entries are case sensitive.

Approximately one minute after the yellow LED stops blinking, check the appropriate Maintenance Tables to confirm that the correct data is displayed for each configured point.

Note: The Carrier Translator polls the CCN system element every 30 seconds to both Read and Write CCN points.

Commissioning the Device on the BACnet/ Modbus Network

The Carrier Translator is now ready to be commissioned on an RS-485 network. This can be done using a BACnet/Modbus front end. Details will depend on the front end product used. Refer to the manufacturer's literature.

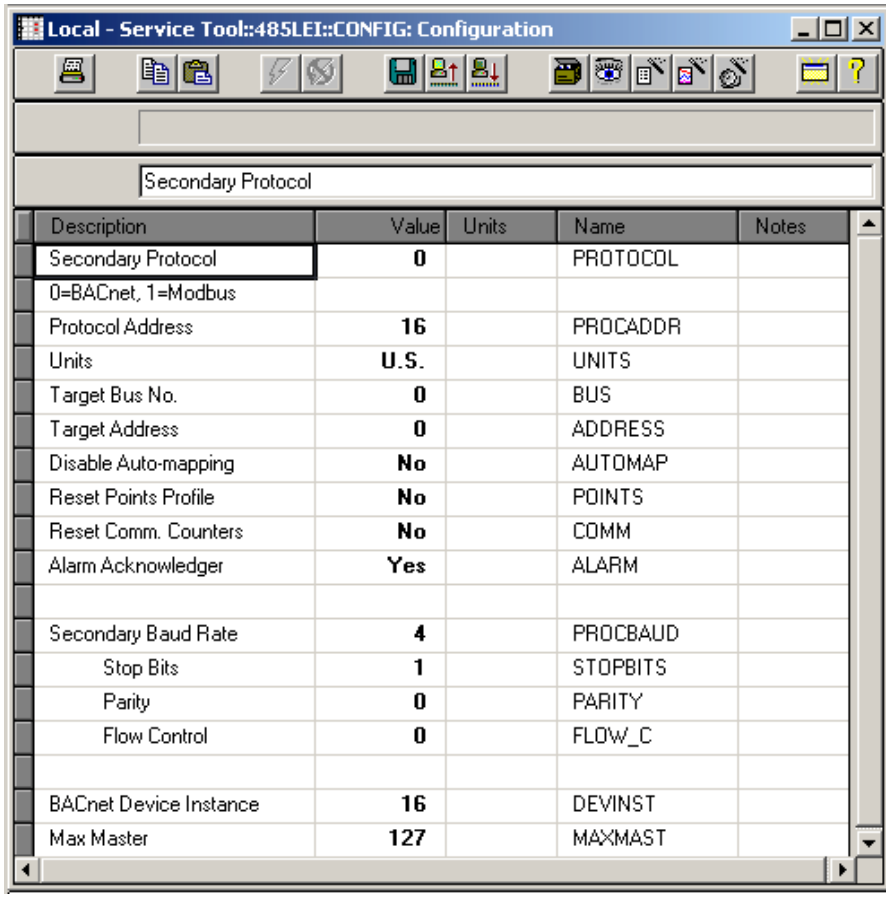
After the BACnet/Modbus Carrier Translator is added to the BACnet/ Modbus network, the BACnet/Modbus network variables that were mapped in this chapter's Map CCN Points with BACnet/Modbus Variables procedure can be used to read/write data points on the target CCN system element.

Carrier Translator Configuration (CONFIG) Table

The Carrier Translator's CONFIG Table is shown below. An explanation of each decision, including allowable entries and default values follows.

Note: When starting up the Carrier Translator this is the first table you must configure. Be sure to coordinate all BACnet/Modbus configuration values with the systems integrator for each site.

Figure 3-1
Carrier Translator Configuration (CONFIG) Table



Description	Value	Units	Name	Notes
Secondary Protocol	0		PROTOCOL	
0=BACnet, 1=Modbus				
Protocol Address	16		PROCADDR	
Units	U.S.		UNITS	
Target Bus No.	0		BUS	
Target Address	0		ADDRESS	
Disable Auto-mapping	No		AUTOMAP	
Reset Points Profile	No		POINTS	
Reset Comm. Counters	No		COMM	
Alarm Acknowledger	Yes		ALARM	
Secondary Baud Rate	4		PROCBAUD	
Stop Bits	1		STOPBITS	
Parity	0		PARITY	
Flow Control	0		FLOW_C	
BACnet Device Instance	16		DEVINST	
Max Master	127		MAXMAST	

Secondary Protocol

Use this decision to select the desired RS-485 protocol, either Modbus or BACnet.

Allowable Entries
0 = BACnet
1 = Modbus

Default Value
0

Protocol Address

Use this decision to specify the address of the Carrier Translator on the RS-485 protocol bus.

Allowable Entries BACnet = 0-127
Modbus = 1-247

Default Value 16

Default Units Setting

Use this decision to specify the RS-485 display units, either US or Metric.

Allowable Entries US/Metric

Default Value US

Target Bus No.

Use this decision to specify the bus number of the CCN system element to which the Carrier Translator is connected.

Note: If you do not enter a value for this decision or if the value is incorrect, the Carrier Translator will attempt to find the element address itself. You will then have to upload this table to check if the correct address has been found.

Allowable Entries 0-239

Default Value 0

Target Address

Use this decision to specify the number of the CCN system element to which the Carrier Translator is connected.

Note: If you do not enter a value for this decision or if the value is incorrect, the Carrier Translator will attempt to find the element address itself. You will then have to upload this table to check if the correct address has been found.

Allowable Entries 0-239

Default Value 0

Disable Auto-mapping

Use this decision to disable or enable the Carrier Translator's automatic mapping feature. Setting this decision to *No* will cause the Carrier Translator to automatically map to the status points in the target CCN system element. You must set this decision to *Yes* if you will be manually configuring the mapping tables.

Allowable Entries No/Yes

Default Value No

Reset Points Profile

Use this decision to clear the contents of all of the Carrier Translator's CCN point to RS-485 variable mapping tables. Note that the Disable Auto-mapping decision must be set to *Yes* for the tables to be cleared. This decision resets to *No* on completion of the operation. The following tables will be cleared:

PT_CFG1 and PT_CFG2
SP_CFG
TS_CFG

Note: Do not save this decision set to *Yes*. Do not cycle Carrier Translator power for at least 30 seconds after downloading this decision set to Yes. If you do cycle power within 30 seconds of downloading this decision as *Yes*, the Carrier Translator will return to its default address of 0,200 at a baud rate of 9600 baud.

Allowable Entries No/Yes

Default Value No

Reset Comm. Counters

Use this decision to clear the COMMSTAT Maintenance Table's Num Successful Messages and Num Failed Messages counters or to clear an EEPROM checksum error from the MESSAGES Maintenance Table. This decision resets to *No* on completion of the operation.

Note: Do not save this decision set to *Yes*.

Allowable Entries No/Yes

Default Value No

Alarm Acknowledger

Use this decision to specify whether the Carrier Translator should act as a CCN alarm acknowledger for all alarm messages received from the CCN Bus. There must be only one CCN alarm acknowledger per CCN.

Note that alarms from the Carrier Translator's target system element are placed into the Carrier Translator's buffer.

Allowable Entries No/Yes

Default Value Yes

Secondary Baud Rate

Use this decision to specify the BACnet/Modbus baud rate.

Allowable Entries
1 = 1200
2 = 2400
3 = 4800
4 = 9600
5 = 19200
6 = 38400

Default Value 4

Stop Bits

Use this decision to specify the number of stop bits, which are extra bits that are used to mark the end of each Modbus communication packet.

This decision applies to Modbus only.

Allowable Entries 0 - 2

Default Value 1

Parity

Parity is an error detection protection routine in which a 0 or 1 bit is added to the end of a group of bits so that it will have an even or odd number of total bits. Use this decision to specify the parity that is used by each Modbus communication packet.

This decision applies to Modbus only.

Allowable Entries
0 = None
1 = Odd
2 = Even

Default Value 0

Flow Control

Flow control is the management of data flow in order that computer devices can handle the exchange of data at an efficient pace. Use this decision to specify the Modbus data flow protocol.

This decision applies to Modbus only.

Allowable Entries
0 = None
1 = Xon/Xoff

Default Value 0

BACnet Device Instance

Use this decision to specify the unique network device identifier for the Carrier Translator in a total BACnet system, which may involve multiple buses. Be sure to coordinate this value with the systems integrator for each site.

This decision applies to BACnet only.

Allowable Entries 1 - 4194303

Default Value 16

Max Master

Use this decision when the Carrier Translator is configured for BACnet protocol to specify the highest protocol address that the Carrier Translator will increment to when polling for the next master. This value is also a writeable property from BACnet.

This decision applies to BACnet only.

Allowable Entries 1 - 127

Default Value 127

Point Mapping Configuration (PT_CFG1 and PT_CFG2) Tables

The Carrier Translator's PT_CFG1 and PT_CFG2 Tables shown in Figure 3-2 and 3-3 are used to map or associate CCN display points with BACnet objects or Modbus registers. Sample PT_CFG1 and PT_CFG2 Tables are shown below. An explanation of each decision follows.

On uploading the PIC tables, the Carrier Translator automatically fills in the PT_CFG1 Table with the first 30 points from the PIC's status display tables and the PT_CFG2 Table with PIC status display table points 31 through 60. The point names in these tables can be modified at any time.

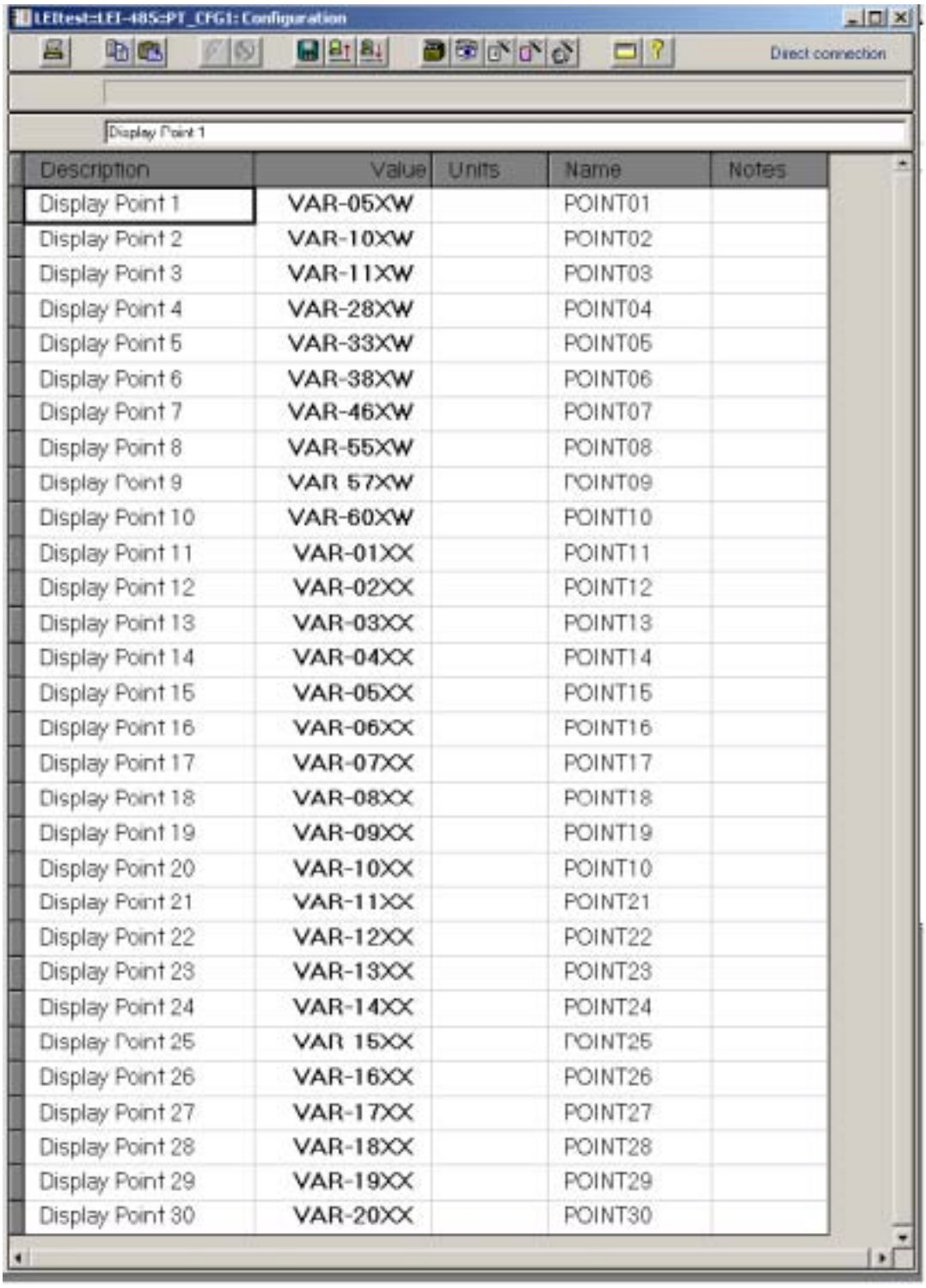
Allowable Entries and Default Value

CCN point names can consist of up to 8 characters. The default for each decision is blank.

Display Point 1 - 60

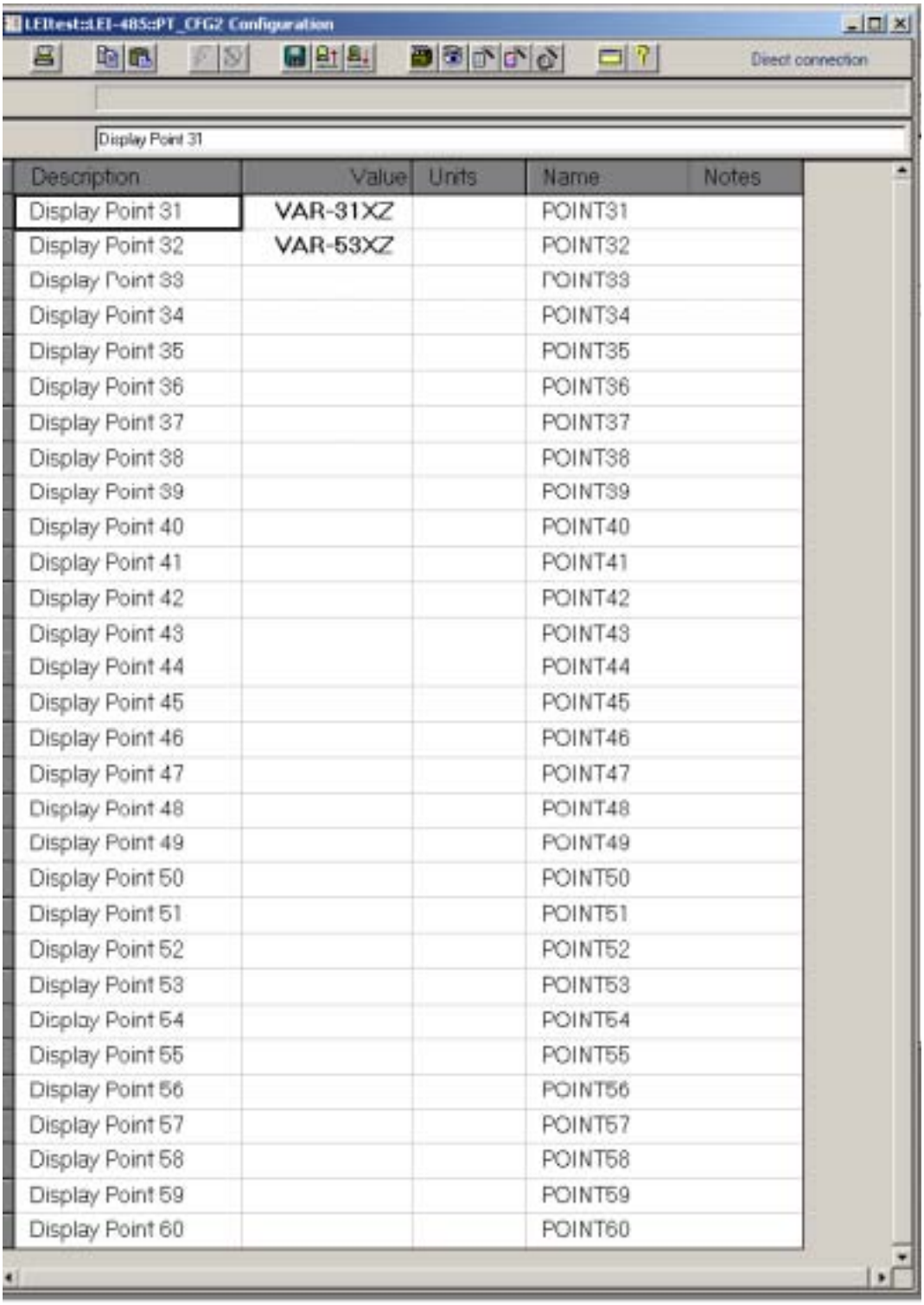
These decisions are used to specify the names of the CCN status display table points that the Modbus or BACnet system can access. Enter the CCN point name in the *Value* field.

Figure 3-2
Sample Point Mapping 1
Configuration (PT_CFG1)
Table



Description	Value	Units	Name	Notes
Display Point 1	VAR-05XW		POINT01	
Display Point 2	VAR-10XW		POINT02	
Display Point 3	VAR-11XW		POINT03	
Display Point 4	VAR-28XW		POINT04	
Display Point 5	VAR-33XW		POINT05	
Display Point 6	VAR-38XW		POINT06	
Display Point 7	VAR-46XW		POINT07	
Display Point 8	VAR-55XW		POINT08	
Display Point 9	VAR-57XW		POINT09	
Display Point 10	VAR-60XW		POINT10	
Display Point 11	VAR-01XX		POINT11	
Display Point 12	VAR-02XX		POINT12	
Display Point 13	VAR-03XX		POINT13	
Display Point 14	VAR-04XX		POINT14	
Display Point 15	VAR-05XX		POINT15	
Display Point 16	VAR-06XX		POINT16	
Display Point 17	VAR-07XX		POINT17	
Display Point 18	VAR-08XX		POINT18	
Display Point 19	VAR-09XX		POINT19	
Display Point 20	VAR-10XX		POINT10	
Display Point 21	VAR-11XX		POINT21	
Display Point 22	VAR-12XX		POINT22	
Display Point 23	VAR-13XX		POINT23	
Display Point 24	VAR-14XX		POINT24	
Display Point 25	VAR-15XX		POINT25	
Display Point 26	VAR-16XX		POINT26	
Display Point 27	VAR-17XX		POINT27	
Display Point 28	VAR-18XX		POINT28	
Display Point 29	VAR-19XX		POINT29	
Display Point 30	VAR-20XX		POINT30	

Figure 3-3
Sample Point Mapping 2
Configuration (PT_CFG2)
Table



Description	Value	Units	Name	Notes
Display Point 31	VAR-31XZ		POINT31	
Display Point 32	VAR-53XZ		POINT32	
Display Point 33			POINT33	
Display Point 34			POINT34	
Display Point 35			POINT35	
Display Point 36			POINT36	
Display Point 37			POINT37	
Display Point 38			POINT38	
Display Point 39			POINT39	
Display Point 40			POINT40	
Display Point 41			POINT41	
Display Point 42			POINT42	
Display Point 43			POINT43	
Display Point 44			POINT44	
Display Point 45			POINT45	
Display Point 46			POINT46	
Display Point 47			POINT47	
Display Point 48			POINT48	
Display Point 49			POINT49	
Display Point 50			POINT50	
Display Point 51			POINT51	
Display Point 52			POINT52	
Display Point 53			POINT53	
Display Point 54			POINT54	
Display Point 55			POINT55	
Display Point 56			POINT56	
Display Point 57			POINT57	
Display Point 58			POINT58	
Display Point 59			POINT59	
Display Point 60			POINT60	

Setpoint Mapping Configuration (SP_CFG) Table

The Carrier Translator's SP_CFG Table is used to map or associate CCN setpoint table points with BACnet objects or Modbus registers. A sample SP_CFG Table is shown below. An explanation of each decision follows.

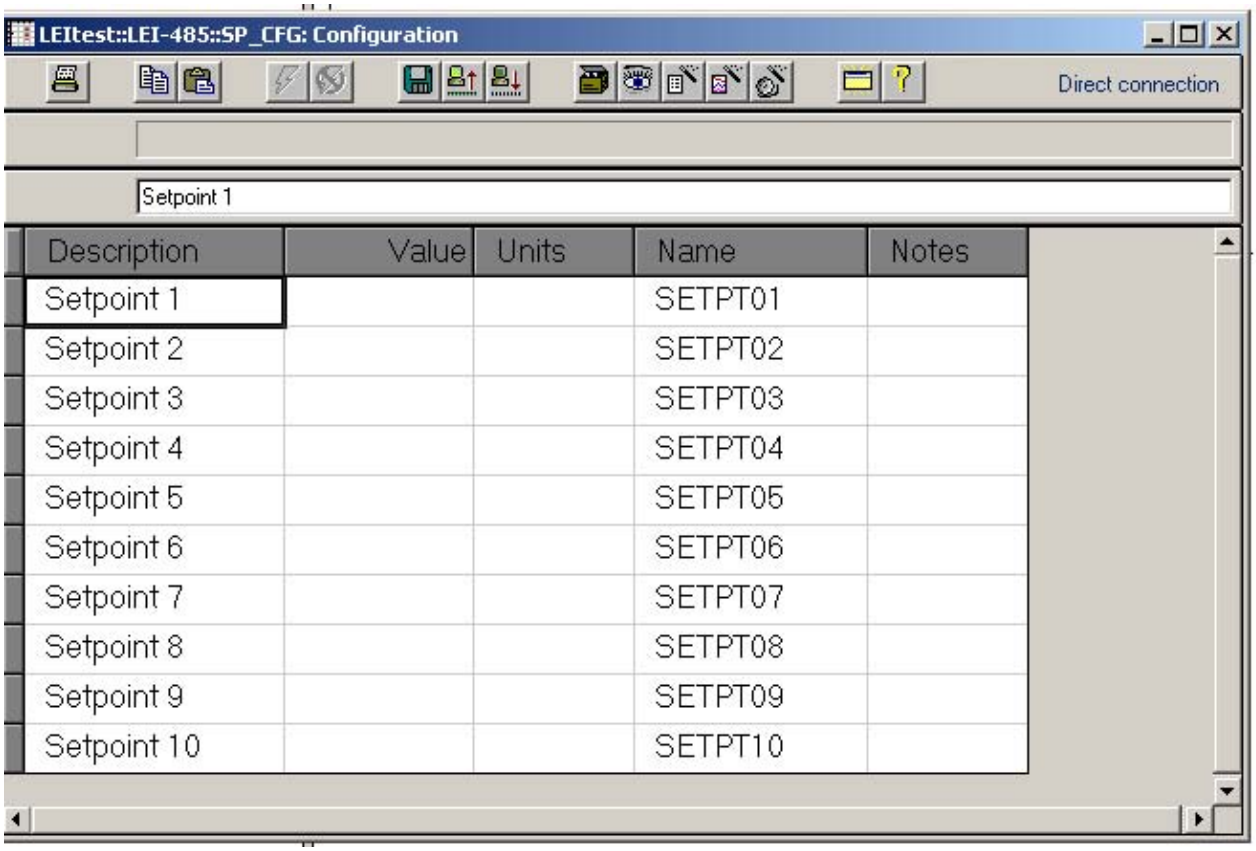
On uploading the PIC tables, the Carrier Translator automatically enters the first 10 point names from the PIC's setpoint table into the SP_CFG Table. The point names can be modified at any time.

Note: If interfacing to a CCN device that contains multiple identical setpoint tables (for example, the Comfort Controller), the Translator will only expose the first instance of a setpoint point name.

Allowable Entries and Default Value

CCN point names can consist of up to 8 characters. The default for each decision is blank.

Figure 3-4
Setpoint Mapping Configuration (SP_CFG) Table



Description	Value	Units	Name	Notes
Setpoint 1			SETPT01	
Setpoint 2			SETPT02	
Setpoint 3			SETPT03	
Setpoint 4			SETPT04	
Setpoint 5			SETPT05	
Setpoint 6			SETPT06	
Setpoint 7			SETPT07	
Setpoint 8			SETPT08	
Setpoint 9			SETPT09	
Setpoint 10			SETPT10	

Setpoint 1 - Setpoint 10 These decisions are used to specify the names of the CCN setpoint table points that the Modbus or BACnet system can access. Enter the CCN setpoint variable name in the *Name* field.

Occupancy Mapping Configuration (TS_CFG) Table

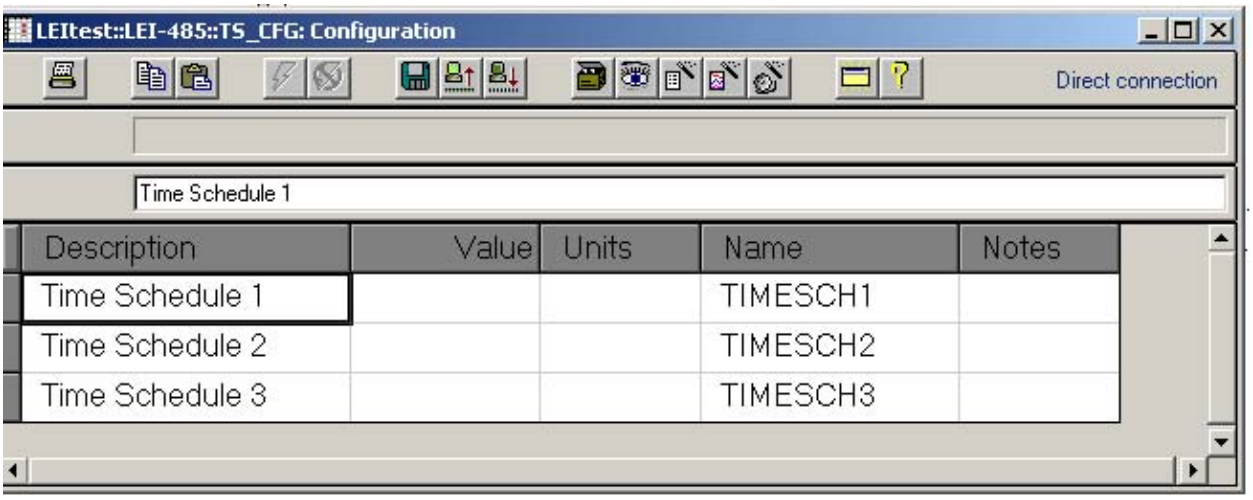
The Carrier Translator's TS_CFG Table is used to map or associate CCN occupancy tables with BACnet objects or Modbus registers. A sample TS_CFG Table is shown below. An explanation of each decision follows.

On uploading the PIC tables, the Carrier Translator automatically fills in TS_CFG Table with the name of the first 3 occupancy tables found in the PIC. The occupancy table names can be modified at any time.

Allowable Entries and Default Value

CCN occupancy table names can consist of up to 8 characters. The default for each decision is blank.

Figure 3-5
Occupancy Mapping Configuration (TS_CFG) Table



Occupancy 1 - Occupancy 3

These decisions are used to specify the names of the CCN occupancy schedules that the BACnet or Modbus system can access. Enter the CCN setpoint variable name in the *Name* field.

Maintenance

Maintenance

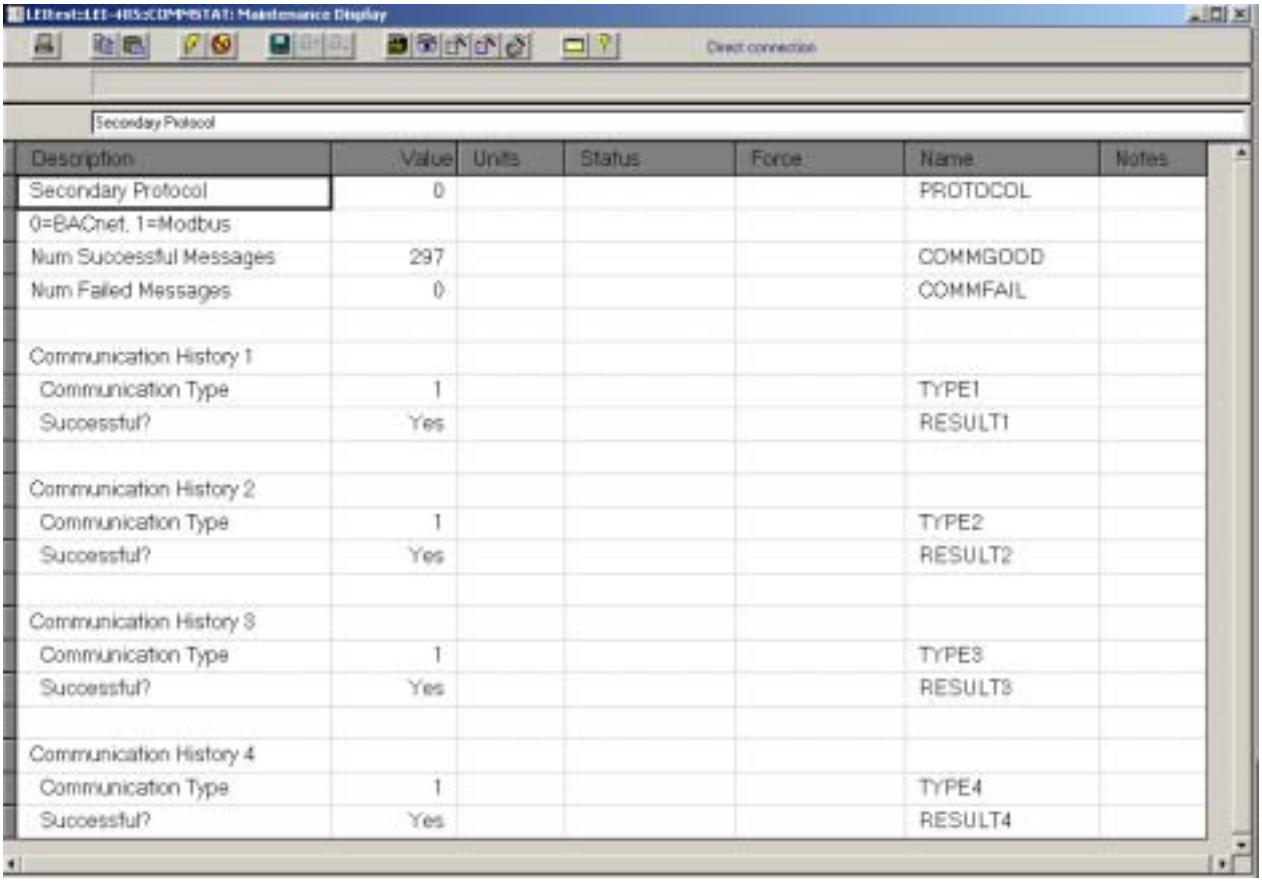
The BACnet/Modbus Carrier Translator's operation contains the following maintenance tables:

COMMSTAT	Communication Status Maintenance Table
MESSAGES	Messages Maintenance Table
PT_MNT1-6	Point Maintenance Tables 1 - 6
SP_MNT	Setpoint Maintenance Table
TS_MNT	Occupancy Maintenance Table

Communication Status Maintenance (COMMSTAT) Table

Figure 4-1 below illustrates the Communication Status Maintenance (COMMSTAT) Table. The maintenance values displayed in this table are read-only values that show diagnostic data about BACnet or Modbus-to-CCN system element communication. These values are updated every 30 seconds. An explanation of each value in the table follows.

Figure 4-1
Communications Status Maintenance (COMMSTAT) Table



Description	Value	Units	Status	Force	Name	Notes
Secondary Protocol	0				PROTOCOL	
0=BACnet, 1=Modbus						
Num Successful Messages	297				COMMGOOD	
Num Failed Messages	0				COMMFAIL	
Communication History 1						
Communication Type	1				TYPE1	
Successful?	Yes				RESULT1	
Communication History 2						
Communication Type	1				TYPE2	
Successful?	Yes				RESULT2	
Communication History 3						
Communication Type	1				TYPE3	
Successful?	Yes				RESULT3	
Communication History 4						
Communication Type	1				TYPE4	
Successful?	Yes				RESULT4	

Secondary Protocol

Indicates the RS-485 protocol, either ModBus or BACnet, as specified in the CONFIG Table's Secondary Protocol configuration decision.

Valid Display
0 = BACnet
1 = Modbus

Num Successful Messages

Indicates the number of successful read, write, and other communication messages sent from ModBus or BACnet to the CCN system element. This value is a running total of the number of successful messages detected since the last communication reset. It is not reset on cycling Carrier Translator power. It can be reset, however, using the CONFIG Table's Reset Comm. Counters decision.

Note: For BACnet, this decision indicates the number of successful application packets, not the number of successful MS/TP packets.

Valid Display 0 - 65,000

Num Failed Messages

Indicates the number of unsuccessful read, write, and other communication messages sent from ModBus or BACnet to the CCN system element. This value is a running total of the number of unsuccessful messages detected since the last communication reset. It is not reset on cycling power to the Carrier Translator. It can be reset, however, using the CONFIG Table's Reset Comm. Counters decision.

Note: For BACnet, this decision indicates the number of failed application packets, not the number of filed MS/TP packets.

Valid Display 0 - 65,000

Communication History 1 - 4

These 4 decisions display the below-listed data about the four most recent BACnet/Modbus-to-CCN communication messages.

Communication Type

Indicates the type of communication message.

Valid Display
1 = Read Network Variable
2 = Write Network Variable
3 = Other

Successful?

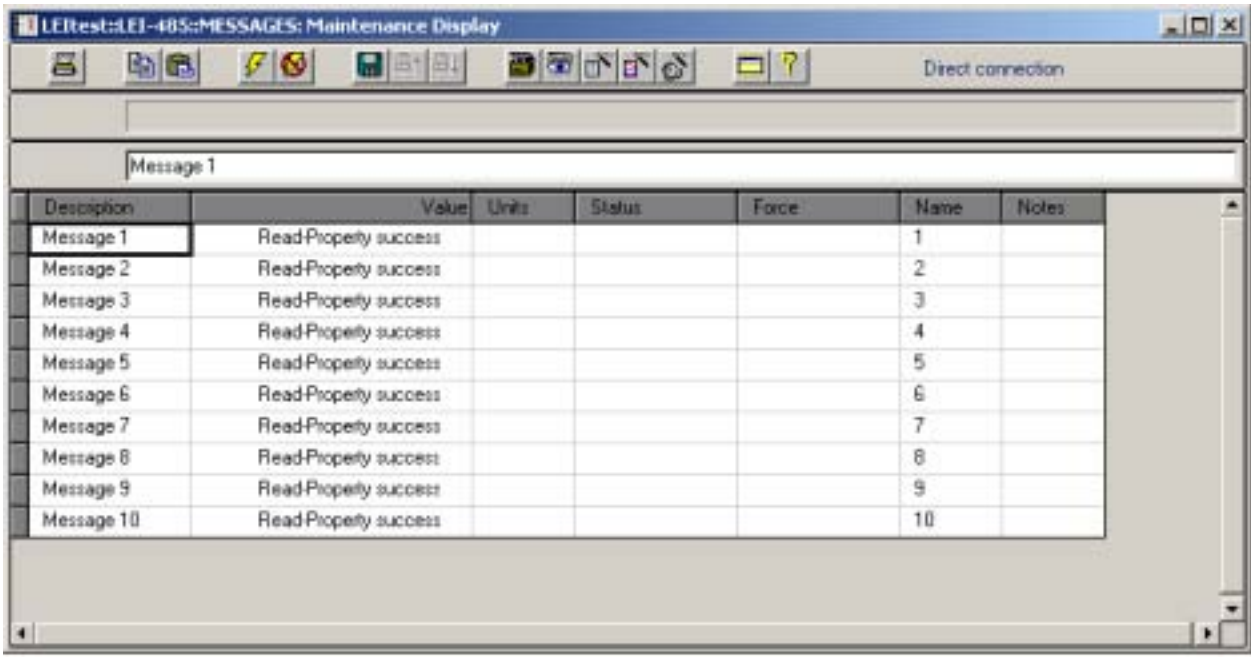
Indicates whether or not this communication was successful.

Valid Display Yes/No

Messages Maintenance (MESSAGES) Table

Figure 4-2 below illustrates the Messages Maintenance (MESSAGES) Table. The maintenance values displayed in this table are read-only values that show diagnostic data about BACnet or Modbus-to-CCN system element communication messages. The table also includes messages relating to the Carrier Translator. These values are updated every 30 seconds. An explanation of each value in the table follows.

Figure 4-2
Messages Maintenance (MESSAGES) Table



Description	Value	Units	Status	Force	Name	Notes
Message 1	Read Property success				1	
Message 2	Read Property success				2	
Message 3	Read Property success				3	
Message 4	Read Property success				4	
Message 5	Read Property success				5	
Message 6	Read Property success				6	
Message 7	Read Property success				7	
Message 8	Read Property success				8	
Message 9	Read Property success				9	
Message 10	Read Property success				10	

Message 1 - 10

These 10 decisions display diagnostic information about the 10 most recent BACnet or Modbus-to-CCN communication messages.

Valid Display

Program Status Messages
LEI POR
Initializing database
Finding controller
Uploading controller
Points configured
Upload failed
Controller not found
Auto-mapped PIC points
Clearing points profile
CCN COMM fail
CCN COMM success
EEPROM checksum error*
Mapped point error

* This message can be cleared by setting the Config Table's Reset Comm. Counters decision to *Yes* and downloading.

BACnet Communication Messages
Transmit IAm
Received alarm
Event-notify fail
Event-notify success
ReadProperty success
ReadProperty fail
WriteProperty success
WriteProperty fail
Get-time
Set-time
Who-has received
Confirmed request resp
Read-file (unsupported)
Write-file (unsupported)
Alarm-ack (unsupported)
Get-alarm (unsupported)
Subscrb-cov (unsupported)
Reset BACnet

Modbus Communication Messages
Read-register success
Read-register fail
Write-register success
Write-register fail

Point Maintenance (PT_MNT) Tables 1-6

Figure 4-4 illustrates the Point Maintenance (PT_MNT1) Table. The maintenance values displayed in these 6 tables are read-only values that show the current value of the CCN points that have been mapped to the BACnet objects or Modbus registers as the points exist in the CCN system element. These values are updated every 30 seconds. An explanation of each value in the table follows.

Display Point 1 - 10

Point Name

These 10 decisions display the point name of the CCN display points that have been mapped or associated with the BACnet objects or Modbus registers. Point mapping is performed in the PT_CFG1 and PT_CFG2 Tables.

Valid Display Up to 8 character CCN point name

Point Value

These 10 decisions display the current value of the CCN display points that have been mapped or associated with the BACnet objects or Modbus registers.

Note: Unmapped points will display a value of -999.99

BACnet Obj/ Modbus Reg

These 10 decisions display the type and instance of the BACnet objects or Modbus registers to which this CCN display point has been associated.

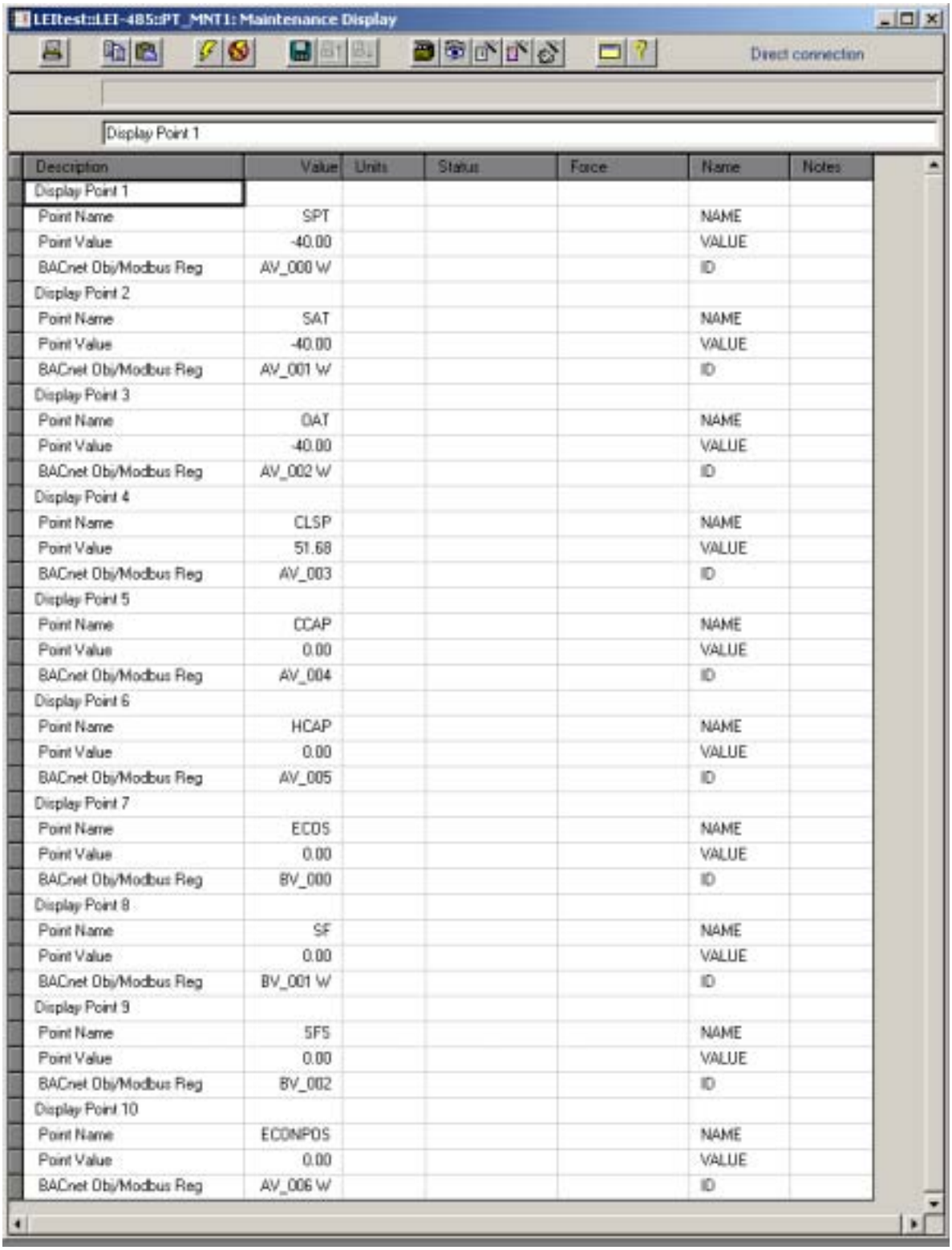
Valid Display Any valid BACnet object or Modbus register

Notes: BACnet object type and instances begin with the characters *AV* or *BV*, where AV=Analog Value and BV=Binary Value, followed by a 3-digit instance number. Ex: AV006. The first instance of each is AV_000 and BV_000.

Modbus registers start at *0x4000* hexadecimal.

A *W* displayed at the end of the field indicates that this point is writable.

Figure 4-4
Point Maintenance
(PT_MNT1)
Table 1



Description	Value	Units	Status	Force	Name	Notes
Display Point 1						
Point Name	SPT				NAME	
Point Value	-40.00				VALUE	
BACnet Obj/Modbus Reg	AV_000 W				ID	
Display Point 2						
Point Name	SAT				NAME	
Point Value	-40.00				VALUE	
BACnet Obj/Modbus Reg	AV_001 W				ID	
Display Point 3						
Point Name	OAT				NAME	
Point Value	-40.00				VALUE	
BACnet Obj/Modbus Reg	AV_002 W				ID	
Display Point 4						
Point Name	CLSP				NAME	
Point Value	51.68				VALUE	
BACnet Obj/Modbus Reg	AV_003				ID	
Display Point 5						
Point Name	CCAP				NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg	AV_004				ID	
Display Point 6						
Point Name	HCAP				NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg	AV_005				ID	
Display Point 7						
Point Name	ECOS				NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg	BV_000				ID	
Display Point 8						
Point Name	SF				NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg	BV_001 W				ID	
Display Point 9						
Point Name	SFS				NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg	BV_002				ID	
Display Point 10						
Point Name	ECONPOS				NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg	AV_006 W				ID	

Setpoint Maintenance (SP_MNT) Table

Figure 4-5 illustrates the Setpoint Maintenance (SP_MNT) Table. The maintenance values displayed in this table are read-only values that show the current value of the first 10 points from the PIC's setpoint table. These CCN setpoints have been mapped to the BACnet objects or Modbus registers in the SP_CFG Table. These values are updated every 30 seconds. An explanation of each value in the table follows.

Setpoint 1 - 10

Point Name

These 10 decisions display the name of the CCN setpoint table points that have been mapped or associated with the BACnet objects or Modbus registers. Setpoint mapping is performed in the SP_CFG Table.

Valid Display Up to 8 character CCN point name

Point Value

These 10 decisions display the current value of the CCN setpoint table points that have been mapped or associated with the BACnet objects or Modbus registers.

Note: Unmapped points will display a value of -999.99

BACnet Obj/ Modbus Reg

These 10 decisions display the type and instance of the BACnet object or Modbus register to which this CCN setpoint has been associated.

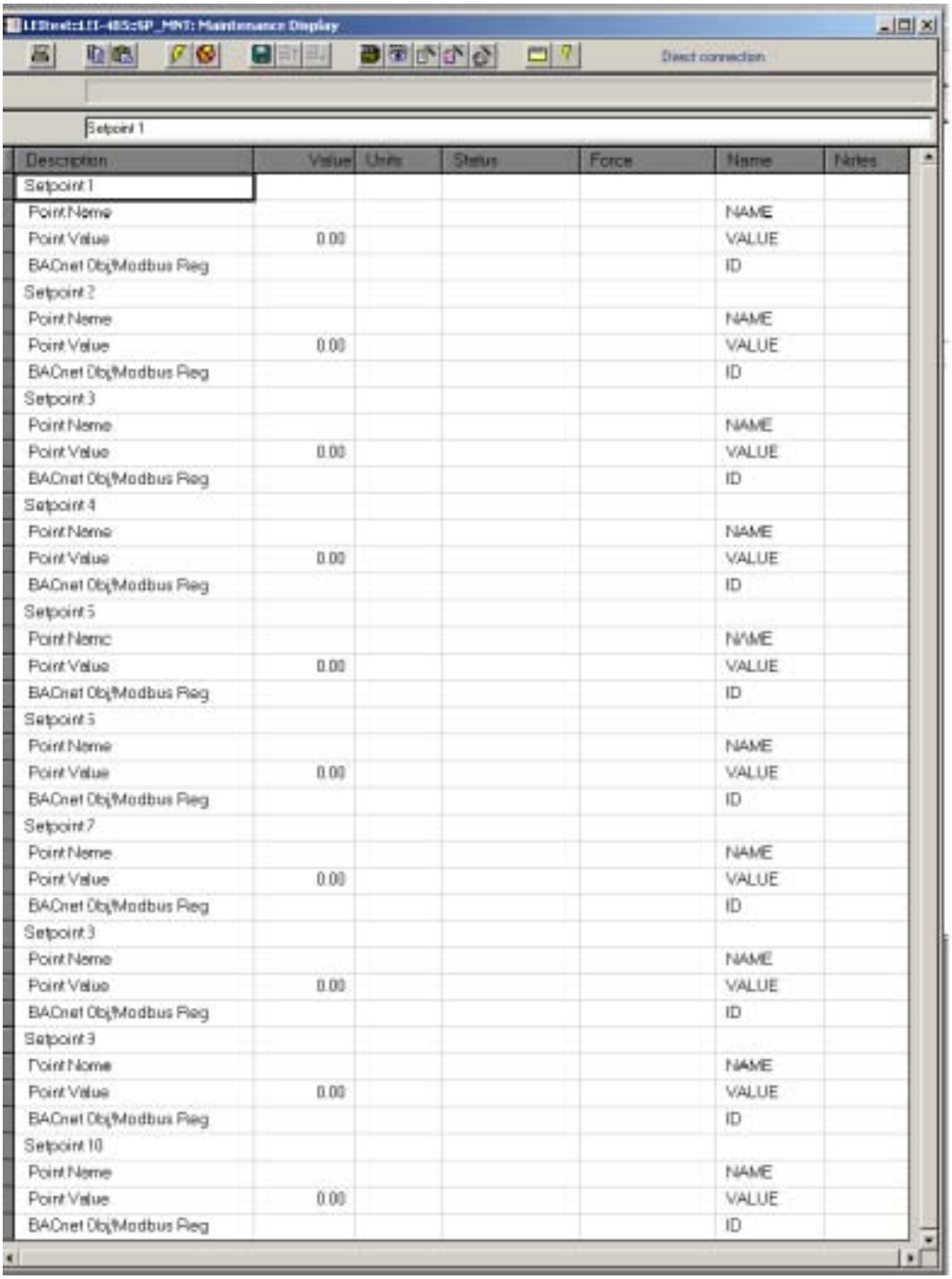
Valid Display Any valid BACnet object or Modbus register

Note: BACnet object type and instance begin with the characters *AV* or *BV*, where *AV*=Analog Value and *BV*=Binary Value, followed by a 3-digit instance number. Ex: AV006

Modbus registers start at *0x8000* hexadecimal.

A *W* displayed at the end of the field indicates that this point is writable.

Figure 4-5
Setpoint Maintenance
(SP_MNT)
Table



Description	Value	Units	Status	Force	Name	Notes
Setpoint 1						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 2						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 3						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 4						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 5						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 6						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 7						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 8						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 9						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	
Setpoint 10						
Point Name					NAME	
Point Value	0.00				VALUE	
BACnet Obj/Modbus Reg					ID	

Occupancy Maintenance (TS_MNT) Table

Figure 4-6 illustrates the Occupancy Maintenance (TS_MNT) Table. The maintenance values displayed in this table are read-only values that show the current value of up to 3 of the PIC's occupancy tables points that have been mapped to the BACnet objects or Modbus registers as the points exist in the CCN system element. These values are updated every 30 seconds. An explanation of each value in the table follows.

Time Schedule 1 - 3

These 3 decisions display the table name of the CCN occupancy tables that have been mapped or associated with the BACnet objects or Modbus registers. Mapping is performed in the TS_CFG Table.

Point Name

Valid Display Up to 8 character CCN table name

Point Value

These 3 decisions indicate whether a CCN occupancy table has been mapped or associated with the BACnet objects or Modbus registers.

Valid Display

1 = CCN table is mapped
0 = CCN table is not mapped

Note: Unmapped points will display a value of -999.99

BACnet Obj/ Modbus Reg

These 3 decisions display the name of the BACnet object or Modbus register to which this CCN occupancy table point has been associated.

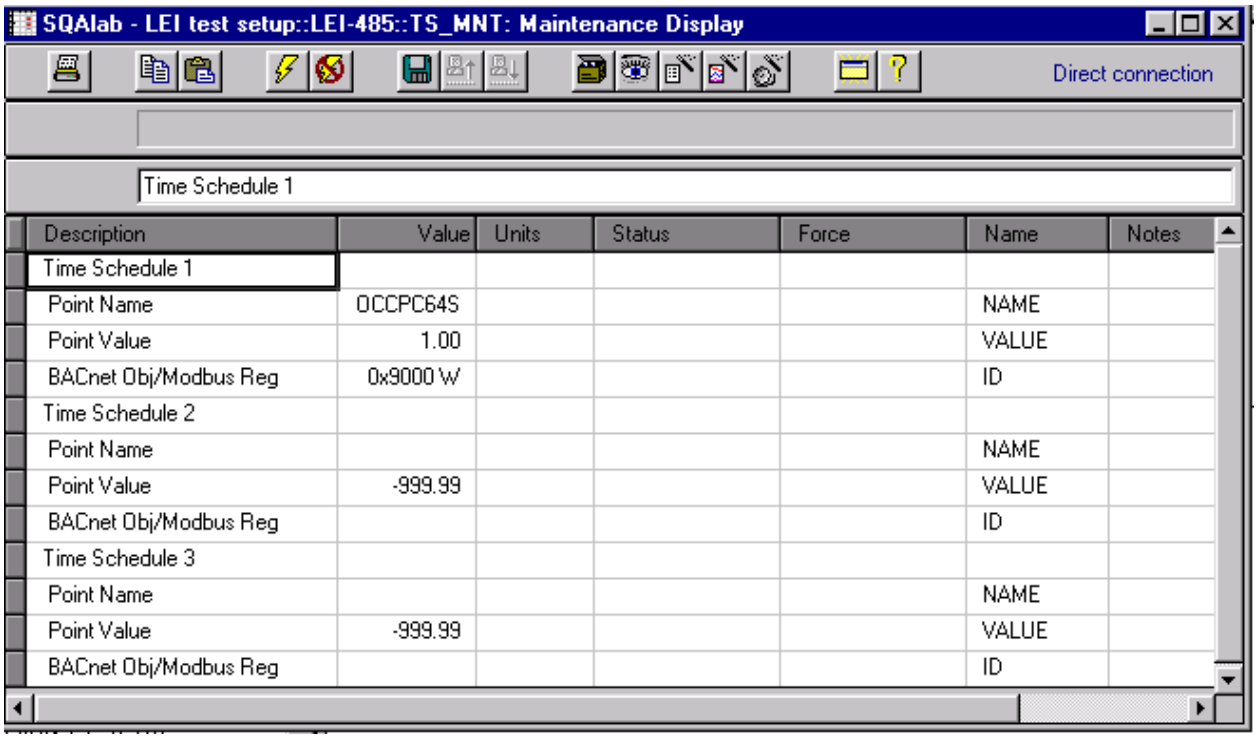
Valid Display Any valid BACnet object or Modbus register

Note: BACnet object names begin with the characters *TS*. The first instance is TS_000.

Modbus registers start at *0x9000* hexadecimal.

A *W* displayed at the end of the field indicates that this point is writable.

Figure 4-6
Occupancy Maintenance
(TS_MNT)
Table



Description	Value	Units	Status	Force	Name	Notes
Time Schedule 1						
Point Name	OCCPC64S				NAME	
Point Value	1.00				VALUE	
BACnet Obj/Modbus Reg	0x9000 W				ID	
Time Schedule 2						
Point Name					NAME	
Point Value	-999.99				VALUE	
BACnet Obj/Modbus Reg					ID	
Time Schedule 3						
Point Name					NAME	
Point Value	-999.99				VALUE	
BACnet Obj/Modbus Reg					ID	

Appendixes

Appendix - A BACnet Point Mapping

This appendix contains the BACnet protocol compliance statement as well as configuration sheets that can be used to indicate BACnet point mapping. These sheets are provided so that they may be photocopied for use as worksheets and hard copy records when configuring the Carrier Translator.

BACnet Protocol Implementation Conformance Statement

Date: January 2005
Vendor Name: CARRIER
Product Name: BACnet Translator
Product Model Number: 33CNTRAN485
Applications Software Version: 1.0 **Firmware Revision:** 1.0 **BACnet Protocol Revision:** 2

Product Description: The BACnet Translator converts data from a single Carrier equipment control (Chiller, RTU or other) into BACnet objects. The object properties are read via a BACnet MSTP data link layer**.**

BACnet Standardized Device Profile (Annex L):

☐ **BACnet Operator Workstation (B-OWS)**
☐ **BACnet Building Controller (B-BC)**
☐ **BACnet Advanced Application Controller (B-AAC)**
■ **BACnet Application Specific Controller (B-ASC) [Note that the Translator is a gateway device]**
☐ **BACnet Smart Sensor (B-SS)**
☐ **BACnet Smart Actuator (B-SA)**

List all BACnet Interoperability Building Blocks Supported (Annex K):

■ K.1.2 BIBB - Data Sharing - ReadProperty-B	(DS-RP-B)
■ K.1.4 BIBB - Data Sharing - ReadPropertyMultiple-B	(DS-RPM-B)
■ K.1.8 BIBB - Data Sharing - WriteProperty-B	(DS-WP-B)
■ K.1.10 BIBB - Data Sharing - WritePropertyMultiple-B	(DS-WPM-B)
■ K.2.2 BIBB – Alarm and Event – Notification Internal-B	(AE-N-I-B)
■ K.5.6 BIBB – Device Management - DeviceCommunicationControl-B	(DM-DCC-B)
■ K.5.1 BIBB - Device Management - Dynamic Device Binding-A	(DM-DDB-A)
■ K.5.2 BIBB - Device Management - Dynamic Device Binding-B	(DM-DDB-B)
■ K.5.3 BIBB - Device Management - Dynamic Object Binding-B	(DM-DOB-A)
■ K.5.12 BIBB - Device Management - TimeSynchronization-B	(DM-TS-B)
■ K.5.16 BIBB – Device Management - ReinitializeDevice-B	(DM-RD-B)

Segmentation Capability: None

☐ Segmented requests supported Window Size _____
☐ Segmented responses supported Window Size _____

Standard Object Types Supported:

Device Object, Analog Value, Binary Value, Schedule and Notification Class Objects

An object type is supported if it may be present in the device. For each standard Object Type supported provide the following data:

1) Whether objects of this type are dynamically creatable using the CreateObject service NO
2) Whether objects of this type are dynamically deletable using the DeleteObject service NO
3) List of the optional properties supported see attached
4) List of all properties that are writable where not otherwise required by this standard see attached
5) List of proprietary properties and for each its property identifier, datatype, and meaning NONE
6) List of any property range restrictions see attached

Data Link Layer Options:

☐ BACnet IP, (Annex J)
☐ BACnet IP, (Annex J), Foreign Device
☐ ISO 8802-3, Ethernet (Clause 7)
☐ ANSI/ATA 878.1, 2.5 Mb. ARCNET (Clause 8)
☐ ANSI/ATA 878.1, RS-485 ARCNET (Clause 8), baud rate(s) _____
■ MS/TP master (Clause 9), baud rate(s): 1200, 2400, 4800, 9600, 19.2K, 38.4K
☐ MS/TP slave (Clause 9), baud rate(s): _____
☐ Point-To-Point, EIA 232 (Clause 10), baud rate(s):_____
☐ Point-To-Point, modem, (Clause 10), baud rate(s): _____
☐ LonTalk, (Clause 11), medium: _____
☐ Other: _____

Device Address Binding:

Is static device binding supported? (This is currently necessary for two-way communication with MS/TP slaves and certain other devices.) ☐Yes ■ No

Networking Options:

☐ Router, Clause 6 - List all routing configurations, e.g., ARCNET-Ethernet, Ethernet-MS/TP, etc.
☐ Annex H, BACnet Tunneling Router over IP
☐ BACnet/IP Broadcast Management Device (BBMD)
Does the BBMD support registrations by Foreign Devices? ☐ Yes ■ No

Character Sets Supported:

Indicating support for multiple character sets does not imply that they can all be supported simultaneously.

■ ANSI X3.4	☐ IBM™/Microsoft™ DBCS	☐ ISO 8859-1
☐ ISO 10646 (UCS-2)	☐ ISO 10646 (UCS-4)	☐ JIS C 6226

If this product is a communication gateway, describe the types of non-BACnet equipment/networks(s) that the gateway supports:

Device (DEV)

ID#	PROPERTY	DEV	Comments
11	APDU_TIMEOUT	REQ	3000
12	APPLICATION_SOFTWARE_VERSION	REQ	Application S/W version from target control
30	DEVICE_ADDRESS_BINDING	REQ	Empty list
44	FIRMWARE_REVISION	REQ	Executive S/W version from target control
62	MAX_APDU_LENGTH_ACCEPTED	REQ	244
70	MODEL_NAME	REQ	BACnet Local Equipment Interface
73	NUMBER_OF_APDU_RETRIES	REQ	3
75	OBJECT_IDENTIFIER	REQ	Object type (8) & instance
76	OBJECT_LIST	REQ	Dependent on CCN controller
77	OBJECT_NAME	REQ	CCN controller name or “RS485/LEI”
79	OBJECT_TYPE	REQ	8
95	PROTOCOL_CONFORMANCE_CLASS	REQ	2
96	PROTOCOL_OBJECT_TYPES_SUPPORTED	REQ	AV, BV, BI, DEV, SCH, NC
97	PROTOCOL_SERVICES_SUPPORTED	REQ	See Annex K above
98	PROTOCOL_VERSION	REQ	1
112	SYSTEM STATUS	REQ	Enum of Operational or Non-Operational
107	SEGMENTATION_SUPPORTED	REQ	Not supported (3)
120	VENDOR_IDENTIFIER	REQ	16
121	VENDOR_NAME	REQ	CARRIER
139	PROTOCOL_REVISION	REQ	2
119	UTC_OFFSET	OPT	0.0
24	DAYLIGHT_SAVINGS_STATUS	OPT	CCN controller DST status
28	DESCRIPTION	OPT	CCN controller description
56	LOCAL_DATE	OPT	CCN controller control date
57	LOCAL_TIME	OPT	CCN controller control time
58	LOCATION	OPT	CCN controller location
63	MAX_INFO_FRAMES	OPT	1
64	MAX_MASTER	R/W	Default of 127

Analog Value (AV)

ID#	PROPERTY		Comments
75	OBJECT_IDENTIFIER	REQ	Object type (2) & instance (0 to 69)
77	OBJECT_NAME	REQ	CCN point name
79	OBJECT_TYPE	REQ	2
85	PRESENT_VALUE	R/W	CCN Point value
28	DESCRIPTION	OPT	CCN point description
117	UNITS	REQ	CCN units mapped to BACnet units
111	STATUS_FLAGS	REQ	CCN status mapped to BACnet status
103	RELIABILITY	OPT	No fault or unreliable other
87	PRIORITY_ARRAY	OPT	CCN force mapped to BACnet priority
104	RELINQUISH_DEFAULT	OPT	0.0
45	HIGH_LIMIT	OPT	High limit of analog
59	LOW_LIMIT	OPT	Low limit of analog
36	EVENT_STATE	REQ	Normal (0)
81	OUT_OF_SERVICE	REQ	1 if point writable, 0 read only
17	NOTIFICATION_CLASS	OPT	Alarm Notification instance (0)

Binary Value (BV)

ID#	PROPERTY		Comments
75	OBJECT_IDENTIFIER	REQ	Object type (5) & instance (0 to 69)
77	OBJECT_NAME	REQ	CCN point name
79	OBJECT_TYPE	REQ	5
85	PRESENT_VALUE	R/W	CCN Point value
28	DESCRIPTION	OPT	CCN point description
117	UNITS	REQ	CCN units mapped to BACnet units
111	STATUS_FLAGS	REQ	CCN status mapped to BACnet status
103	RELIABILITY	OPT	No fault or unreliable other
4	ACTIVE_TEXT	OPT	CCN On text for discrete
46	INACTIVE_TEXT	OPT	CCN Off text for discrete
87	PRIORITY_ARRAY	OPT	CCN force mapped to BACnet priority
104	RELINQUISH_DEFAULT	OPT	0
36	EVENT_STATE	REQ	Normal (0)
81	OUT_OF_SERVICE	REQ	1 if point writable, 0 read only
17	NOTIFICATION_CLASS	OPT	Alarm Notification instance (0)

Binary Input (BI)

ID#	PROPERTY		Comments
75	OBJECT_IDENTIFIER	REQ	Object type (3) & instance N (0 to 3)
77	OBJECT_NAME	REQ	OCCSTATE _N or ALARMSTATE
79	OBJECT_TYPE	REQ	3
28	DESCRIPTION	OPT	“Occupancy State N” or CCN Alarm
85	PRESENT_VALUE	R/W	1 if Alarm or Schedule mapped, else 0
81	OUT_OF_SERVICE	REQ	Read only (0)
84	POLARITY	REQ	Normal (0)

Schedule (SCH)

ID#	PROPERTY		Comments
75	OBJECT_IDENTIFIER	REQ	Object type (17) & instance N (0 to 2)
77	OBJECT_NAME	REQ	CCN schedule name
79	OBJECT_TYPE	REQ	17
28	DESCRIPTION	OPT	"Occupancy Schedule N"
32	EFFECTIVE_PERIOD	REQ	Undefined (0xFF's)
85	PRESENT_VALUE	REQ	1 = mapped, 0 = unmapped
123	WEEKLY_SCHEDULE	R/W	CCN TS mapped to BACnet schedule
88	PRIORITY_FOR_WRITING	REQ	16
54	LIST_OF_OBJECT_PROPERTY_REFERENCES	REQ	BI object, instance N

Notification Class (NC)

ID#	PROPERTY		Comments
75	OBJECT_IDENTIFIER	REQ	Object type (15) & instance (0)
77	OBJECT_NAME	REQ	ALARMNOTIFICATION
79	OBJECT_TYPE	REQ	15
28	DESCRIPTION	OPT	"CCN Alarms"
17	NOTIFICATION_CLASS	REQ	0
86	PRIORITY	REQ	Array of 0,0,0
1	ACK_REQUIRED	REQ	Bit string of 000
102	RECIPIENT_LIST	R/W	Writable from BACnet

BACnet Point Mapping

AV = Analog Value
BV = Binary Value

Object Type (AV/BV)	Instance	Name	Description	Write	Units

Object Type (AV/BV)	Instance	Name	Description	Write	Units

Object Type (AV/BV)	Instance	Name	Description	Write	Units

Object Type (AV/BV)	Instance	Name	Description	Write	Units

Appendix - B Modbus Point Mapping

This appendix contains the Modbus protocol compliance statement, a description of ModScan32, which is Modbus test software that you can download free of charge, as well as tables and configuration sheets that can be used for Modbus point mapping. The configuration sheets are provided so that they may be photocopied for use as worksheets and hard copy records when configuring the Carrier Translator.

Modbus Protocol Implementation Compliance Statement

Modbus Communication

Supported Mode:	RTU
Transmission Media:	EIA-485 at 9600, 19200, and 38400 baud
Message Bits:	8
Parity:	None, Odd, Even
Stop Bits:	0,1,2
Flow Control:	None, XON/XOFF
Modbus Slave Address:	1 to 247

Modbus RTU Supported Function Codes

Function Code	Function
03	Read Multiple Holding Registers
04	Read Multiple Input Registers
06	Write Single Register
16	Write Multiple Registers

ModScan32 Modbus Test Software

ModScan32 is a Microsoft® Windows® application which operates as a Modbus master. It allows you to access and change data points in a connected slave device using the RTU Transmission mode. ModScan is ideally suited for quick and easy compliance testing of the modbus protocol and its built-in display of serial traffic allows effective trouble-shooting of field connections. Visit *www.win-tech.com* for more information on ModScan32 and to download a free ModScan32 trial demo.

Modbus Information Registers (Decimal: 1 – 3) (Hexadecimal: 0x1 – 3)

Modbus Register (HEX)	Object Type	Description	Write
1 (0x01)	Integer	Modbus Address	Yes
2 (0x02)	Integer	Modbus Baud Rate	Yes
3 (0x03)	Integer	Units (0) US, (1) Metric	Yes

PIC Information Registers (Decimal: 4096 – 4098) (Hexadecimal: 0x1000 – 1003)

Modbus Register (HEX)	Object Type	Description	Write
4096 (0x1000)	Integer	PIC Bus Number	No
4097 (0x1001)	Integer	PIC Address Number	No
4098 (0x1002)	Integer	PIC Baud Rate	No

Time and Date Registers (Decimal: 8192 – 8194) (Hexadecimal: 0x2000 – 2002)

Modbus Register (HEX)	Object Type	Description	Write
8192 (0x2000)	Integer	Hour	Yes Yes
8193 (0x2001)	Integer	Minutes	Yes Yes
8194 (0x2002)	Integer	Bits 0-3: DOW	Yes
8194 (0x2002)	Integer	Bit 4 : Daylight Savings ON	Yes
8194 (0x2002)	Integer	Bit 5 : Daylight Savings OFF	Yes
8194 (0x2002)	Integer	Bit 6 : Today is a Holiday	Yes
8194 (0x2002)	Integer	Bit 7 : Tomorrow is a Holiday	Yes

Configuration Sheets

The pages which follow contain configuration sheets that may be photocopied for use as worksheets and hard copy records when mapping Modbus to Carrier Translator points.

The following sheets are provided:

- Point Value Registers
- Setpoint Registers
- Time Schedule Registers

Point Value Registers (Decimal: 16384 – 16443) (Hexadecimal: 0x4000 – 403B)
The values in these registers are truncated integer representations of the actual point values. If the point value is a floating type containing decimal places, the value in these registers will show only the truncated integer portion of the number. For example, if an actual point value that is mapped to Register 0x4000 is 72.5, Register 0x4000 will represent that value as 72 and Register 0x4100 will represent that same value as 725.

Point Value Registers (Decimal: 16640 – 16699) (Hexadecimal: 0x4100 – 413B)
The registers shown on the pages which follow are mirror registers of the Point Value Registers 0x4000 – 0X403B Hexadecimal shown on the previous page. Data in the registers on the pages which follow is multiplied by a factor of 10. For example, a register value of 725 represents an actual point value of 72.5. A discrete value of 10 represents an actual value of 1. In the event where a point value is greater then 3276 a register overflow will occur. For points that have an upper range greater then 3276 a corresponding 0x40xx register should be used.

Point Value Registers (Decimal: 16896 – 16955) (Hexadecimal: 0x4200 – 423B)
The registers shown on the pages which follow are mirror registers of the Point Value Registers 0x4000 – 0X403B Hexadecimal shown earlier. Data in the registers on the pages which follow is multiplied by a factor of 100. For example, a register value of 7250 represents an actual point value of 72.5. A discrete value of 100 represents an actual value of 1. In the event where a point value is greater then 327.6 a register overflow will occur. For points that have an upper range greater then 327.6 a corresponding 0x41xx register should be used.

Point Value Registers (Decimal: 17152– 17211) (Hexadecimal: 0x4300 – 433B)
The registers shown on the pages which follow are mirror registers of the Point Value Registers 0x4000 – 0X403B Hexadecimal shown earlier. Data in the registers on the pages which follow is multiplied by a factor of 1000. For example, 72500 would repersent 72.5, but this value will be clamped at 32767. In this example, a corresponding 0x42xx register should be used. (A discrete value of 1000 represents an actual value of 1.)

Point Value Registers

Modbus Registers (HEX)				Description	Write	Point Name	Units
X1	X10	X100	X1000				
16384 (0x4000)	16640 (0x4100)	16896 (0x4200)	17152 (0x4300)				
16385 (0x4001)	16641 (0x4101)	16897 (0x4201)	17153 (0x4301)				
16386 (0x4002)	16642 (0x4102)	16898 (0x4202)	17154 (0x4302)				
16387 (0x4003)	16643 (0x4103)	16899 (0x4203)	17155 (0x4303)				
16388 (0x4004)	16644 (0x4104)	16900 (0x4204)	17156 (0x4304)				
16389 (0x4005)	16645 (0x4105)	16901 (0x4205)	17157 (0x4305)				
16390 (0x4006)	16646 (0x4106)	16902 (0x4206)	17158 (0x4306)				
16391 (0x4007)	16647 (0x4107)	16903 (0x4207)	17159 (0x4307)				
16392 (0x4008)	16648 (0x4108)	16904 (0x4208)	17160 (0x4308)				
16393 (0x4009)	16649 (0x4109)	16905 (0x4209)	17161 (0x4309)				
16394 (0x400A)	16650 (0x410A)	16906 (0x420A)	17162 (0x430A)				
16395 (0x400B)	16651 (0x410B)	16907 (0x420B)	17163 (0x430B)				
16396 (0x400C)	16652 (0x410C)	16908 (0x420C)	17164 (0x430C)				
16397 (0x400D)	16653 (0x410D)	16909 (0x420D)	17165 (0x430D)				
16398 (0x400E)	16654 (0x410E)	16910 (0x420E)	17166 (0x430E)				
16399 (0x400F)	16655 (0x410F)	16911 (0x420F)	17167 (0x430F)				
16400 (0x4010)	16656 (0x4110)	16912 (0x4210)	17168 (0x4310)				
16401 (0x4011)	16657 (0x4111)	16913 (0x4211)	17169 (0x4311)				
16402 (0x4012)	16658 (0x4112)	16914 (0x4212)	17170 (0x4312)				
16403 (0x4013)	16659 (0x4113)	16915 (0x4213)	17171 (0x4313)				
16404 (0x4014)	16660 (0x4114)	16916 (0x4214)	17172 (0x4314)				
16405 (0x4015)	16661 (0x4115)	16917 (0x4215)	17173 (0x4315)				
16406 (0x4016)	16662 (0x4116)	16918 (0x4216)	17174 (0x4316)				

16407 (0x4017)	16663 (0x4117)	16919 (0x4217)	17175 (0x4317)				
16408 (0x4018)	16664 (0x4118)	16920 (0x4218)	17176 (0x4318)				
16409 (0x4019)	16665 (0x4119)	16921 (0x4219)	17177 (0x4319)				
16410 (0x401A)	16666 (0x411A)	16922 (0x421A)	17178 (0x431A)				
16411 (0x401B)	16667 (0x411B)	16923 (0x421B)	17179 (0x431B)				
16412 (0x401C)	16668 (0x411C)	16924 (0x421C)	17180 (0x431C)				
16413 (0x401D)	16669 (0x411D)	16925 (0x421D)	17181 (0x431D)				
16414 (0x401E)	16670 (0x411E)	16926 (0x421E)	17182 (0x431E)				
16415 (0x401F)	16671 (0x411F)	16927 (0x421F)	17183 (0x431F)				
16416 (0x4020)	16672 (0x4120)	16928 (0x4220)	17184 (0x4320)				
16417 (0x4021)	16673 (0x4121)	16929 (0x4221)	17185 (0x4321)				
16418 (0x4022)	16674 (0x4122)	16930 (0x4222)	17186 (0x4322)				
16419 (0x4023)	16675 (0x4123)	16931 (0x4223)	17187 (0x4323)				
16420 (0x4024)	16676 (0x4124)	16932 (0x4224)	17188 (0x4324)				
16421 (0x4025)	16677 (0x4125)	16933 (0x4225)	17189 (0x4325)				
16422 (0x4026)	16678 (0x4126)	16934 (0x4226)	17190 (0x4326)				
16423 (0x4027)	16679 (0x4127)	16935 (0x4227)	17191 (0x4327)				
16424 (0x4028)	16680 (0x4128)	16936 (0x4228)	17192 (0x4328)				
16425 (0x4029)	16681 (0x4129)	16937 (0x4229)	17193 (0x4329)				
16426 (0x402A)	16682 (0x412A)	16938 (0x422A)	17194 (0x432A)				
16427 (0x402B)	16683 (0x412B)	16939 (0x422B)	17195 (0x432B)				
16428 (0x402C)	16684 (0x412C)	16940 (0x422C)	17196 (0x432C)				
16429 (0x402D)	16685 (0x412D)	16941 (0x422D)	17197 (0x432D)				
16430 (0x402E)	16686 (0x412E)	16942 (0x422E)	17198 (0x432E)				

16431 (0x402F)	16687 (0x412F)	16943 (0x422F)	17199 (0x432F)				
16432 (0x4030)	16688 (0x4130)	16944 (0x4230)	17200 (0x4330)				
16433 (0x4031)	16689 (0x4131)	16945 (0x4231)	17201 (0x4331)				
16434 (0x4032)	16690 (0x4132)	16946 (0x4232)	17202 (0x4332)				
16435 (0x4033)	16691 (0x4133)	16947 (0x4233)	17203 (0x4333)				
16436 (0x4034)	16692 (0x4134)	16948 (0x4234)	17204 (0x4334)				
16437 (0x4035)	16693 (0x4135)	16949 (0x4235)	17205 (0x4335)				
16438 (0x4036)	16694 (0x4136)	16950 (0x4236)	17206 (0x4336)				
16439 (0x4037)	16695 (0x4137)	16951 (0x4237)	17207 (0x4337)				
16440 (0x4038)	16696 (0x4138)	16952 (0x4238)	17208 (0x4338)				
16441 (0x4039)	16697 (0x4139)	16953 (0x4239)	17209 (0x4339)				
16442 (0x403A)	16698 (0x413A)	16954 (0x423A)	17210 (0x433A)				
16443 (0x403B)	16699 (0x413B)	16955 (0x423B)	17211 (0x433B)				

Status Registers (Hexadecimal: 0x5000 – 0x503B), (Decimal: 20480 – 20539)
Status registers show current alarm status and force priority of a CCN point. Each status register corresponds to a point value register. Point value registers begin at 0x4000 and end at 0x403B hexadecimal (16384 - 16443 decimal). The status of a CCN point that is mapped to a 0x4000 hex register is shown in a 0x5000 hex register.

Value Register (Hex)	Status Register (Hex)
0x4000	0x5000
0x4001	0x5001
0x4002	0x5003
:	:
0x403B	0x503B

Each status register is an 8-bit value where the low order nibble (bits 0 – 3) represents the alarm status and the high order nibble (bits 4 – 7) represents the force priority for the corresponding point.

Example: (8-bits) 0111 0011 = 115
High Order Nibble, Low Order Nibble

Force Priority = 7 Alarm Status = 3

Alarm Status - Low Order Nibble (bits 0 – 3):

0 = error
1 = hardware/communication error
2 = software error
3 = low alarm limit exceeded or discrete alarm
4 = high alarm limit exceeded

Force Priority - High Order Nibble (bits 4 – 7):

0 = No force in effect
1 = Fire force in effect
2 = Safety force in effect
3 = Service force in effect
4 = Supervisor force in effect
5 = Monitor / Remote force in effect
6 = Min-off time force in effect
7 = Controlling force in effect
8 = BEST force in effect
9 = Temperature force in effect
10 = Loadshed force in effect

In the example shown on the previous page, a value of 115 in a status register represents a point that has a Controlling Force in effect as well as a Low Alarm Limit alarm.

To remove a force, write a value of 0 to the status register of that point.

Setpoint Registers (Decimal: 32768 – 32777) (Hexadecimal: 0x8000 – 8009)
The values in these registers are truncated integer representations of the actual point values. If the setpoint value is a floating type containing decimal places, the value in these registers will show only the truncated integer portion of the number. For example, if an actual setpoint value that is mapped to Register 0x4000 is 72.5, Register 0x4000 will represent that value as 72 and Register 0x4100 will represent that same value as 725.

Setpoint Registers (Decimal: 33024 – 33033) (Hexadecimal: 0x8100 – 8109)
Data in the registers shown on the next page is multiplied by a factor of 10. For example a register value of 725 represents an actual point value of 72.5. A discrete value of 10 represents an actual value of 1. In the event where a point value is greater then 3276 a register overflow will occur.

Setpoint Registers (Decimal: 33280 – 33289) (Hexadecimal: 0x8200 – 8209)
Data in the registers shown on the next page is multiplied by a factor of 100. For example, a register value of 7250 represents an actual point value of 72.5. A discrete value of 100 represents an actual value of 1. In the event where a point value is greater then 327.6 a register overflow will occur. For points that have an upper range greater then 327.6 a corresponding 0x41xx register should be used.

Setpoint Registers (Decimal: 33536 – 33545) (Hexadecimal: 0x8300 – 8309)
Data in the registersshown on the next page is multiplied by a factor of 1000. For example, 72500 would repersent 72.5, but this value will be clamped at 32767. (In this example, a corresponding 0x42xx register should be used.) A discrete value of 1000 represents an actual value of 1.

Setpoint Registers

Modbus Registers (HEX)				Description	Write	Point Name	Units
X1	X10	X100	X1000				
32768 (0x8000)	33024 (0x8100)	33280 (0x8200)	33536 (0x8300)				
32769 (0x8001)	33025 (0x8101)	33281 (0x8201)	33537 (0x8301)				
32770 (0x8002)	33026 (0x8102)	33282 (0x8202)	33538 (0x8302)				
32771 (0x8003)	33027 (0x8103)	33283 (0x8203)	33539 (0x8303)				
32772 (0x8004)	33028 (0x8104)	33284 (0x8204)	33540 (0x8304)				
32773 (0x8005)	33029 (0x8105)	33285 (0x8205)	33541 (0x8305)				
32774 (0x8006)	33030 (0x8106)	33286 (0x8206)	33542 (0x8306)				
32775 (0x8007)	33031 (0x8107)	33287 (0x8207)	33543 (0x8307)				
32776 (0x8008)	33032 (0x8108)	33288 (0x8208)	33544 (0x8308)				
32777 (0x8009)	33033 (0x8109)	33289 (0x8209)	33545 (0x8309)				

Time Schedule Registers
Decimal: 36864-36888 Hexadecimal: 0x9000-9018
Time Schedule #1

Modbus Register (HEX)	Object Type	Description	Write	Point Name	Units
36864 (0x9000)	Integer	Manual Override Hours	Yes	OVRD	hours
36865 (0x9001)	Integer	Period 1: Day of Week	Yes	DOW1	
36866 (0x9002)	Integer	Period 1: Occupied from	Yes	OCC1	
36867 (0x9003)	Integer	Period 1: Occupied to	Yes	UNOCC1	
36868 (0x9004)	Integer	Period 2: Day of Week	Yes	DOW2	
36869 (0x9005)	Integer	Period 2: Occupied from	Yes	OCC2	
36870 (0x9006)	Integer	Period 2: Occupied to	Yes	UNOCC2	
36871 (0x9007)	Integer	Period 3: Day of Week	Yes	DOW3	
36872 (0x9008)	Integer	Period 3: Occupied from	Yes	OCC3	
36873 (0x9009)	Integer	Period 3: Occupied to	Yes	UNOCC3	
36874 (0x900A)	Integer	Period 4: Day of Week	Yes	DOW4	
36875 (0x900B)	Integer	Period 4: Occupied from	Yes	OCC4	
36876 (0x900C)	Integer	Period 4: Occupied to	Yes	UNOCC4	
36877 (0x900D)	Integer	Period 5: Day of Week	Yes	DOW5	
36878 (0x900E)	Integer	Period 5: Occupied from	Yes	OCC5	
36879 (0x900F)	Integer	Period 5: Occupied to	Yes	UNOCC5	
36880 (0x9010)	Integer	Period 6: Day of Week	Yes	DOW6	
36881 (0x9011)	Integer	Period 6: Occupied from	Yes	OCC6	
36882 (0x9012)	Integer	Period 6: Occupied to	Yes	UNOCC6	

Modbus Register (HEX)	Object Type	Description	Write	Point Name	Units
36883 (0x9013)	Integer	Period 7: Day of Week	Yes	DOW7	
36884 (0x9014)	Integer	Period 7: Occupied from	Yes	OCC7	
36885 (0x9015)	Integer	Period 7: Occupied to	Yes	UNOCC7	
36886 (0x9016)	Integer	Period 8: Day of Week	Yes	DOW8	
36887 (0x9017)	Integer	Period 8: Occupied from	Yes	OCC8	
36888 (0x9018)	Integer	Period 8: Occupied to	Yes	UNOCC8	

Time Schedule Registers
Decimal: 36889-36913 Hexadecimal: 0x9019-9031
Time Schedule #2

Modbus Register (HEX)	Object Type	Description	Write	Point Name	Units
36889 (0x9019)	Integer	Manual Override Hours	Yes	OVRD	hours
36890 (0x901A)	Integer	Period 1: Day of Week	Yes	DOW1	
36891 (0x901B)	Integer	Period 1: Occupied from	Yes	OCC1	
36892 (0x901C)	Integer	Period 1: Occupied to	Yes	UNOCC1	
36893 (0x901D)	Integer	Period 2: Day of Week	Yes	DOW2	
36894 (0x901E)	Integer	Period 2: Occupied from	Yes	OCC2	
36895 (0x901F)	Integer	Period 2: Occupied to	Yes	UNOCC2	
36896 (0x9020)	Integer	Period 3: Day of Week	Yes	DOW3	
36897 (0x9021)	Integer	Period 3: Occupied from	Yes	OCC3	
36898 (0x9022)	Integer	Period 3: Occupied to	Yes	UNOCC3	
36899 (0x9023)	Integer	Period 4: Day of Week	Yes	DOW4	
36900 (0x9024)	Integer	Period 4: Occupied from	Yes	OCC4	
36901 (0x9025)	Integer	Period 4: Occupied to	Yes	UNOCC4	
36902 (0x9026)	Integer	Period 5: Day of Week	Yes	DOW5	
36903 (0x9027)	Integer	Period 5: Occupied from	Yes	OCC5	
36904 (0x9028)	Integer	Period 5: Occupied to	Yes	UNOCC5	
36905 (0x9029)	Integer	Period 6: Day of Week	Yes	DOW6	
36906 (0x902A)	Integer	Period 6: Occupied from	Yes	OCC6	
36907 (0x902B)	Integer	Period 6: Occupied to	Yes	UNOCC6	

Modbus Register (HEX)	Object Type	Description	Write	Point Name	Units
36908 (0x902C)	Integer	Period 7: Day of Week	Yes	DOW7	
36909 (0x902D)	Integer	Period 7: Occupied from	Yes	OCC7	
36910 (0x902E)	Integer	Period 7: Occupied to	Yes	UNOCC7	
36911 (0x902F)	Integer	Period 8: Day of Week	Yes	DOW8	
36912 (0x9030)	Integer	Period 8: Occupied from	Yes	OCC8	
36913 (0x9031)	Integer	Period 8: Occupied to	Yes	UNOCC8	

Time Schedule Registers
Decimal: 36914-36938 Hexadecimal: 0x9032-904A
Time Schedule #3

Modbus Register (HEX)	Object Type	Description	Write	Point Name	Units
36914 (0x9032)	Integer	Manual Override Hours	Yes	OVRD	hours
36915 (0x9033)	Integer	Period 1: Day of Week	Yes	DOW1	
36916 (0x9034)	Integer	Period 1: Occupied from	Yes	OCC1	
36917 (0x9035)	Integer	Period 1: Occupied to	Yes	UNOCC1	
36918 (0x9036)	Integer	Period 2: Day of Week	Yes	DOW2	
36919 (0x9037)	Integer	Period 2: Occupied from	Yes	OCC2	
36920 (0x9038)	Integer	Period 2: Occupied to	Yes	UNOCC2	
36921 (0x9039)	Integer	Period 3: Day of Week	Yes	DOW3	
36922 (0x903A)	Integer	Period 3: Occupied from	Yes	OCC3	
36923 (0x903B)	Integer	Period 3: Occupied to	Yes	UNOCC3	
36924 (0x903C)	Integer	Period 4: Day of Week	Yes	DOW4	
36925 (0x903D)	Integer	Period 4: Occupied from	Yes	OCC4	
36926 (0x903E)	Integer	Period 4: Occupied to	Yes	UNOCC4	
36927 (0x903F)	Integer	Period 5: Day of Week	Yes	DOW5	
36928 (0x9040)	Integer	Period 5: Occupied from	Yes	OCC5	
36929 (0x9041)	Integer	Period 5: Occupied to	Yes	UNOCC5	
36930 (0x9042)	Integer	Period 6: Day of Week	Yes	DOW6	
36931 (0x9043)	Integer	Period 6: Occupied from	Yes	OCC6	
36932 (0x9044)	Integer	Period 6: Occupied to	Yes	UNOCC6	

Modbus Register (HEX)	Object Type	Description	Write	Point Name	Units
36933 (0x9045)	Integer	Period 7: Day of Week	Yes	DOW7	
36934 (0x9046)	Integer	Period 7: Occupied from	Yes	OCC7	
36935 (0x9047)	Integer	Period 7: Occupied to	Yes	UNOCC7	
36936 (0x9048)	Integer	Period 8: Day of Week	Yes	DOW8	
36937 (0x9049)	Integer	Period 8: Occupied from	Yes	OCC8	
36938 (0x904A)	Integer	Period 8: Occupied to	Yes	UNOCC8	

Index

Index