A Series® Lighting Control Panelboards

A Series® Lighting Controller Modbus Register Map

Introduction

The Lighting Controller can be configured as a Modbus Slave device using the RTU (Remote Terminal Unit) Modbus Master-Slave protocol. The Lighting Controller is always the slave and cannot initiate communications.

MODBUS devices communicate using a master-slave technique in which only one device (the master) can initiate transactions (called queries). The other devices (slaves) respond by supplying the requested data to the master, or by taking the action requested in the query. A slave is any peripheral device (I/O transducer, valve, network drive, or other measuring device), which processes information and sends its output to the master using MODBUS. Masters can address individual slaves, or can initiate a broadcast message to all slaves. Slaves return a response to all queries addressed to them individually, but do not respond to broadcast queries

MODBUS devices usually include a Register Map. MODBUS functions operate on Register map registers to monitor, configure, and control module I/O. You should refer to the Register map for your device to gain a better understanding of its operation.

The transmission mode defines the bit contents of the message bytes transmitted along the network, and how the message information is to be packed into the message stream and decoded.

Standard MODBUS networks employ one of two types of transmission modes:

i) ASCII Mode ii) RTU Mode.

The mode of transmission is usually selected along with other serial port communication parameters (baud rate, parity, etc.) as part of the device configuration.

RTU Transmission Mode

In RTU (Remote Terminal Unit) Mode, each 8-bit message byte contains two 4-bit Hexadecimal characters, and the message is transmitted in a continuous stream. The greater effective character density increases throughput over ASCII mode at the same baud rate.

Message frame

A message frame is used to mark the beginning and ending point of a message allowing the receiving device to determine which device is being addressed and to know when the message is completed. It also allows partial messages to be detected and errors flagged as a result.

A MODBUS message is placed in a message frame by the transmitting device. Each word of this message (including the frame) is also placed in a data frame that appends a start bit, stop bit, and parity bit.

In ASCII mode, the word size is 7 bits, while in RTU mode; the word size is 8 bits. Thus, every 8 bits of an RTU message is effectively 11 bits when accounting for the start, stop, and parity bits of the data frame.

RTU Mode Message Frames

RTU mode messages start with a silent interval of at least 3.5 character times. Implemented as a multiple of character times at the baud rate being used on the network. The first field transmitted is the device address. The allowable characters transmitted for all fields are hexadecimal values 0-9, A-F. A networked device continuously monitors the network, including the silent intervals, and when the first field is received (the address) after a silent interval of at least 3.5 character times, the device decodes it to determine if it is the addressed device. Following the last character transmitted, a similar silent interval of 3.5 character times marks the end of the message and a new message can begin after this interval.

The entire message must be transmitted as a continuous stream. If a silent interval of more than 1.5 character times occurs before completion of the frame (not a continuous stream), the receiving device flushes the incomplete message and assumes the next byte will be the address field of a new message.

In similar fashion, if a new message begins earlier than 3.5 character times following a

previous message, the receiving device assumes it is a continuation of the previous message. This will generate an error, as the value in the final CRC field will not be valid for the combined messages.

Message Format

The Modbus RTU Protocol is strictly based upon a transaction scheme where a master device generates a query and a slave device replies with a response. Each query and response message transaction consists of the following four parts

MODBUS ADDRESSES

The master device addresses a specific slave device by placing the 8-bit slave address in the address field of the message (RTU Mode). The address field of the message frame contains two characters (in ASCII mode), or 8 binary bits (in RTU Mode). Valid addresses are from 1-247.

When the slave responds, it places its own address in this field of its response to let the master know which slave is responding.

MODBUS FUNCTIONS

The function code field of the message frame will contain two characters (in ASCII mode), or 8 binary bits (in RTU Mode) that tell the slave what kind of action to take. Valid function codes are from 1-255, but not all codes will apply to a module and some codes are reserved for future use.

MODBUS DATA FIELD

The data field provides the slave with any additional information required by the slave to complete the action specified by the function code. The data is formed from a multiple of character bytes (a pair of ASCII characters in ASCII Mode), or a multiple of two hex digits in RTU mode, in range 00H-FFH. The data field typically includes register addresses; count values, and

written data. If no error occurs, the data field of a response from a slave will return the requested data. If an error occurs, the data field returns an exception code that the master's application software can use to determine the next action to take.

CRC Error Checking

RTU Mode message frames include an error checking method that is based on a Cyclical Redundancy Check (CRC). The error-checking field of a message frame contains a 16-bit value (two 8-bit bytes) that contains the result of a Cyclical Redundancy Check (CRC) calculation performed on the message contents.

The checksum field lets the receiving device determine if a packet is corrupted with transmission errors. In Modbus RTU mode, a 16-bit Cyclic Redundancy Check (CRC-16) is used. The sending device calculates a 16-bit value, based on every byte in the packet, using the CRC-16 algorithm. The calculated value is inserted in the error check field.

The receiving device performs the calculation, without the error check field; on the entire packet it receives. The resulting value is compared to the error check field.

Transmission errors occur when the calculated checksum is not equal to the Checksum stored in the incoming packet. The receiving device ignores a bad packet.

Function Codes

The Lighting Controller operating as a Modbus slave uses the function codes listed in Table 1.

Diagnostics

Diagnostics are supported by the Modbus protocol with the following three methods:

- *Exception Responses* – When a failure occurs a diagnostic value is returned, otherwise an acknowledgement is returned.
- *Read Exception Status* – Hardware-specific coil values that can be read and interpreted.
- *Function 08* –Formal layout of reporting statistics and problems.

The Lighting Controller only supports exception responses. These occur when a message is received by the Modbus slave, but the device was unable to answer due to a problem with the message, such as a request for a register that does not exist. Thorough reporting of problems is important for development, support, and use of the product. There are eight exception codes that are provided by the Modbus protocol. The remaining values in the exception byte can be assigned arbitrarily.

If the slave receives a message without a communication error, but is unable to process it, the slave returns an exception response to inform the master of the error. The high-order bit is set in the function code and returned with the characterizing exception response code, as listed in Table2.

Function Code	Command	Purpose	Register Group
01	Read Coil Status	Read results or status of coils	Command coils, binary outputs
02	Read Input Status	Read physical binary inputs	Input status
03	Read Holding Registers	Reading the setpoint registers	Fixed value registers, setpoint registers
04	Read Input Registers	Reading actual value registers	Schedule event registers and feedback inputs
05	Force Single Coil	Setting the signal coil or executing single command	Command coils, binary outputs
06	Preset Single Register	Write data into register	Setpoint registers
15	Force Multiple Coils	Write data to multiple outputs	Binary outputs
16	Write Multiple Registers	Write data into multiple register	Setpoint registers, etc.

Table 1. Function codes supported by the Lighting Controller.

Function	Name	Purpose
2	Invalid register	The Register address is not recognized or defined by the Lighting Controller.
3	Invalid data value	Data out of range
83h	Partial register read	Partial property read
84h	Partial register write	Partial property write
85h	Write protect violation	Write to a read only property.
86h	Single write entry violation	Multiple entries on single write packet, which is restricted to one entry modification due to array shuffling. NOTE: Only for LG lighting outputs and common group properties.
89h	Invalid property read/write	Fail to read/write a property due to restriction.

Table 2. Exception response codes.

Objects Supported by the Lighting Controller

The Lighting Controller supports the object types listed in **Error! Reference source not found.**.

Register Types

All registers are defined as read only except coil and setpoint registers. Fixed-value register addresses start at 0201 and are read by function code 04. Actual-value register addresses start at 1000 and are read by function code 03. Event register addresses start at 20001 and are read by function code 04.

All registers consist of 16-bit integers, with values less than 256 stored in the lower byte. Inputs, outputs, and binary values use Modbus coils and input bit order.

The Modbus device is mapped to a BACnet database.

The Modbus object ID for BACnet object type is given in Table4.

Object	Quantity	Comments
DEV – Device Information	1 per controller	Fixed registers
INPUTS:		
MI – Multi-state Input	80 = 16 inputs * 5 Modules	Switch Input Modules.
AI - Analog Input	19 = 3 On-board + (8 inputs * 2 Modules)	Daylight Optimization Modules.
AV – Analog Variables	16 = 8 inputs * 2 Modules	Daylight Optimization Modules.
OUTPUTS:		
BO - Binary Output	66 = Breakers 4 = Special Outputs	
BV – Binary Variables	66	1 Light switch per Breaker
Operational Control:		
LG - Lighting Group	16	Each controlling up to 33 local outputs.
SCH – Schedule	16	7 Day schedules - with 16 On/Off times per day (+ 2 Calendar references)
CAL – Calendar	2	90 Date entries spanning 10 years
CEL - Compact Event Log	1 per controller	1500 timestamp event records
LS – Load Shed	1	66 outputs, 5 shed levels

Table 3. Object types supported by the Lighting Controller.

Object Type	Object ID for MODBUS
AI - Analog Input	0
AO - Analog Output	1
AV - Analog Variable	2
BI - Binary Input	3
BO - Binary Output	4
BV - Binary Variable	5
CAL – Calendar	6
DEV – Device Information Object	8
CO – Control Loop	12
MI – Multi-state Input	13
MO – Multi-state Output	14
MV – Multi-state Value	19
SCH – Schedule	17
TL - Trend Log	20
AIC - Analog Input Configuration	176
BDC - Binary Device Configuration	178
MIC – Multi-state Input Configuration	183
LS – Load Shed	182
CEL – Compact Event Log	297
LNK – LINKnet Device	299
LG - Lighting Group	303

Table 4. Object ID

Fixed-Value Registers

Fixed-value register addresses start at 10001 and are read by Function Code 04. They are mapped to Device properties. These properties are assigned constant values at the time of manufacture and may be stored in the onboard EEPROM rather than in the database.

The fixed-value (device object property) registers are listed in Table 5.

All strings have an embedded length. The first two registers of the string contain the length of the string; e.g., Product ID Reg10040 Hi contains the most-significant byte and Reg10040 Lo contains the least-significant byte of the string length. The first character starts at Reg10041 Hi, the second character in Reg10041 Lo, the third character in Reg10042 Hi, and so forth.

Offset	Modbus # reg	Property	Format	Range	Type
0	10001–10007	Firmware Version	ASCII	Int (ASCII string) 12 chars max + 2 bytes string size	R
7	10008–10014	Hardware Revision	ASCII	Int (ASCII string) 12 chars max + 2 bytes string size	R
14	10015–10021	Application software version	ASCII	Int (ASCII string) 12 chars max	R
21	10022–10023	Latitude[1]	Real	±90.0	R
23	10024–10025	Longitude[2]	Real	±180.0	R
25	10026	UTC Enable	Int	0 /1	R
26	10027	UTC Offset	Int	±32767	R
27	10028	DST Enable	Int	0/1	R
Last Reset Time/Date					
28	10029	Hours/Minutes	Word		R
29	10030	Seconds/Hundreths sec	Word		R
30	10031	DayOfWeek/Day	Word		R
31	10032	Month/Year	Word		R
32	10033	Reset Count	Word	1–65536	R
33	10034	Device Address (Mac address)	Word	1–99	R
34	10035	Device Control	Word	See Reg 50035	NA
Current Time/Date					
35	10036	Hours/Minutes	Word		R
36	10037	Seconds/Hundredths Sec	Word		R
37	10038	DayofWeek/Day	Word		R
38	10039	Month/Year	Word		R
39	10040–10046	Product ID	ASCII	INT (ASCII string) 12 chars max + 2 bytes string size	R

1 Latitude: positive value is north, negative value is south.
2 Longitude: positive value is east, negative value is west.

Table 5. Types of fixed-value registers.

Date and time values are represented as follows:

- Week of day: 1–7, 255 (wildcard)
- Day: 1–31, 255 (wildcard)
- Month: 1–12, 255 (wildcard)
- Year: 0–255
- Hour: 0–23, 255 (wildcard)
- Minutes: 0–59, 255 (wildcard)
- Seconds: 0–59, 255 (wildcard)
- Hundredths of seconds: 0–99, 255 (wildcard)

Sample Packets

The following are examples of packets generated by requests for specific properties.

Device Product ID

Table 6 shows the result of a request for device product ID: Reg10040 to Reg10046 (Function 04, DEV1 in map).

The response is as follows:

Reg10040 – Product Id Length – 0x000A – 10 characters

Reg10041 to Reg10046 – Product Id – 0x444C432B473136343245 – "DLC+G1642E"

All unused bytes are filled with zeros.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	27	Byte Count	0E
Register Address Lo	37	Reg10040 Hi	00
Quantity of Input Reg. Hi	00	Reg10040 Lo	0A
Quantity of Input Reg. Lo	07	Reg10041 Hi	44
Packet CRC Hi		Reg10041 Lo	4C
Packet CRC Lo		Reg10042 Hi	43
		Reg10042 Lo	2B
		Reg10043 Hi	47
		Reg10043 Lo	31
		Reg10044 Hi	36
		Reg10044 Lo	34
		Reg10045 Hi	32
		Reg10045 Lo	45
		Reg10046 Hi	00
		Reg10046 Lo	00
		Packet CRC Hi	
		Packet CRC Lo	

Table 6. Sample packet for device product ID.

Latitude and Longitude

Table 7 shows the result of a request for device latitude and longitude: Reg10022–Reg10025 (Function 04, DEV1 in map).

The response is as follows:

Reg10022 to Reg10023 – Latitude – 0x4226CCCD – 41.7 degrees North

Reg10024 to Reg10025 – Longitude – 0x428FCCCD – 71.9 degrees West (–71.9)

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	27	Byte Count	08
Register Address Lo	25	Reg10022 Hi	42
Quantity of Input Reg. Hi	00	Reg10022 Lo	26
Quantity of Input Reg. Lo	04	Reg10023 Hi	CC
Packet CRC Hi		Reg10023 Lo	CD
Packet CRC Lo		Reg10024 Hi	C2
		Reg10024 Lo	8F
		Reg10025 Hi	CC
		Reg10025 Lo	CD
		Packet CRC Hi	
		Packet CRC Lo	

Table 7. Sample packet for device latitude and longitude.

Last Reset

Table 8 shows the result of a request for device last reset time and date and reset count: Reg10029–Reg10033 (Function 04, DEV1 in map).

The response is as follows:

Reg10029 to Reg10030 – Last Reset Time: 0x0E080400 – 14:08:04.00

Reg10031 to Reg10032 – Last Reset Date: 0x 04040368 – Thursday Mar 4, 2004

Reg10033 – Reset Count – 0x0005 – 5

- Our base year is 1900 ie. 0x68 (104)

  Year = 1900+104=2004

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	27	Byte Count	0A
Register Address Lo	2C	Reg10029 Hi	0E
Quantity of Input Reg. Hi	00	Reg10029 Lo	08
Quantity of Input Reg. Lo	05	Reg10030 Hi	04
Packet CRC Hi		Reg10030 Lo	00
Packet CRC Lo		Reg10031 Hi	04
		Reg10031 Lo	04
		Reg10032 Hi	03
		Reg10032 Lo	68
		Reg10033 Hi	00
		Reg10033 Lo	05
		Packet CRC Hi	
		Packet CRC Lo	

Table 8. Sample packet for device last reset date, time, and count.

Device Properties (Editable)

Table 9 indicates the editable device properties.

Offset	Modbus # reg	Property	Format	Range	Type
0	50001 – 50007	Firmware Version	ASCII	Int (ASCII string) 12 chars max + 2 bytes string size	R
7	50008 – 50014	Hardware Revision	ASCII	Int (ASCII string) 12 chars max + 2 bytes string size	R
14	50015 – 50021	Application software version	ASCII	Int (ASCII string) 12 chars max	R
21	50022 – 50023	Latitude*	Real	+/- 90.0	R/W
23	50024 – 50025	Longitude**	Real	+/- 180.0	R/W
25	50026	UTC Enable	INT	0 /1	R/W
26	50027	UTC Offset	INT	+/- 32767	R/W
27	50028	DST Enable	INT	0/1	R/W
Last Reset Time/Date					
28	50029	Hours/Minutes	WORD		R
29	50030	Seconds/Hundreths sec	WORD		R
30	50031	DayOfWeek/Day	WORD		R
31	50032	Month/Year	WORD		R
32	50033	Reset Count	WORD	1 – 65535	R
33	50034	Device Address (Mac address)	WORD	1 – 99	R
34	50035	Device Control	WORD	1 - Load-from-Flash 2 - Save-to-Flash 3 - Recfg Network	W
Current Time/Date					
35	50036	Hours/Minutes	WORD		R/W
36	50037	Seconds/Hundredths Sec	WORD		R/W
37	50038	DayofWeek/Day	WORD		R/W
38	50039	Month/Year	WORD		R/W
39	50040 – 50046	Product Id	ASCII	INT (ASCII string) 12 chars max + 2 bytes string size	R

Table 9. Types of fixed-value registers.

Note:

- Registers of Type R must not be written to even though the can be viewed with Function 3.
- Registers of Type R/W are designed to be editable. This allows setting the DST/UTC & Location.
- Changing the Current time requires a single write to registers 50035 to 50038, using Function 16

Multistate Input Registers

What would normally be binary inputs for the switch inputs on the Lighting Controller are treated as multistate inputs because of the different types of switches that can be applied to these devices (LinkNet). The value is an enumerated type, which has more than two possible values. These are read only, so function 4 is used to obtain the analog values that represent the switch states.

Mapped to multistate inputs (MI) 101–116, 201–216, 301–316, 401–416, and 501–516. The Modbus address space is a contiguous block starting at 1. The value in the register represents the input's PresentValue.

Multistate Input Registers

The multistate input registers are listed in Table10. A sample display of 16 switch inputs is shown in Figure1. Enumerated inputs can have the following values:

- Off (Hex 01) • On (Hex 02) • N/A (Hex 03)

Reg.	Contents	Type	Format	Object Instance
1	MI101 Value	R	Enumerated	101
2	MI102 Value	R	Enumerated	102
3	MI103 Value	R	Enumerated	103
4	MI104 Value	R	Enumerated	104
5	MI105 Value	R	Enumerated	105
6	MI106 Value	R	Enumerated	106
7	MI107 Value	R	Enumerated	107
8	MI108 Value	R	Enumerated	108
9	MI109 Value	R	Enumerated	109
10	MI110 Value	R	Enumerated	110
11	MI111 Value	R	Enumerated	111
12	MI112 Value	R	Enumerated	112
13	MI113 Value	R	Enumerated	113
14	MI114 Value	R	Enumerated	114
15	MI115 Value	R	Enumerated	115
16	MI116 Value	R	Enumerated	116
17	MI201 Value	R	Enumerated	201
18	MI202 Value	R	Enumerated	202
19	MI203 Value	R	Enumerated	203
20	MI204 Value	R	Enumerated	204
21	MI205 Value	R	Enumerated	205
22	MI206 Value	R	Enumerated	206
23	MI207 Value	R	Enumerated	207
24	MI208 Value	R	Enumerated	208
25	MI209 Value	R	Enumerated	209
26	MI210 Value	R	Enumerated	210
27	MI211 Value	R	Enumerated	211
28	MI212 Value	R	Enumerated	212
29	MI213 Value	R	Enumerated	213
30	MI214 Value	R	Enumerated	214
31	MI215 Value	R	Enumerated	215
32	MI216 Value	R	Enumerated	216
33	MI301 Value	R	Enumerated	301
34	MI302 Value	R	Enumerated	302
35	MI303 Value	R	Enumerated	303
36	MI304 Value	R	Enumerated	304
37	MI305 Value	R	Enumerated	305
38	MI306 Value	R	Enumerated	306
39	MI307 Value	R	Enumerated	307
40	MI308 Value	R	Enumerated	308
41	MI309 Value	R	Enumerated	309
42	MI310 Value	R	Enumerated	310
43	MI311 Value	R	Enumerated	311
44	MI312 Value	R	Enumerated	312
45	MI313 Value	R	Enumerated	313
46	MI314 Value	R	Enumerated	314
47	MI315 Value	R	Enumerated	315
48	MI316 Value	R	Enumerated	316
49	MI401 Value	R	Enumerated	401
50	MI402 Value	R	Enumerated	402
51	MI403 Value	R	Enumerated	403
52	MI404 Value	R	Enumerated	404
53	MI405 Value	R	Enumerated	405
54	MI406 Value	R	Enumerated	406
55	MI407 Value	R	Enumerated	407
56	MI408 Value	R	Enumerated	408
57	MI409 Value	R	Enumerated	409
58	MI410 Value	R	Enumerated	410
59	MI411 Value	R	Enumerated	411
60	MI412 Value	R	Enumerated	412
61	MI413 Value	R	Enumerated	413
62	MI414 Value	R	Enumerated	414
63	MI415 Value	R	Enumerated	415
64	MI416 Value	R	Enumerated	416
65	MI501 Value	R	Enumerated	501
66	MI502 Value	R	Enumerated	502
67	Reserved	-	-	503
68	Reserved	-	-	504
69	Reserved	-		505
70	Reserved	-	-	506
71	Reserved	-	-	507
72	Reserved	-	-	508
73	Reserved	-	-	509
74	Reserved	-	-	510
75	Reserved	-	-	511
76	Reserved	-	-	512
77	Reserved	-	-	513
78	Reserved	-	-	514
79	Reserved	-	-	515
80	Reserved	-	-	516

Table 10. List of physical input registers.

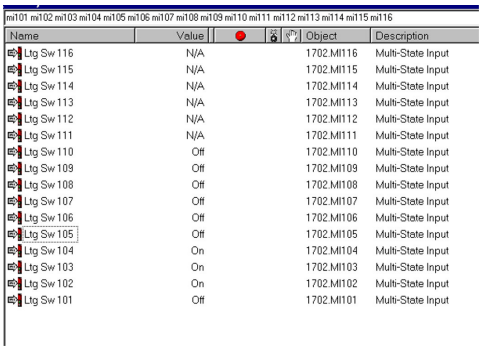

Figure 1. Sample display of 16 switch inputs.

Sample Packets

The following are examples of packets generated by requests for specific groups of inputs.

Read MI Reg1 to Reg16

Table 11 shows the result of a request to read MI Reg1 to Reg16 (Function 4, MI 101–116 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	00	Byte Count	20
Register Address Lo	00	Input 101 Hi	00
Quantity of Input Reg. Hi	00	Input 101 Lo	01
Quantity of Input Reg. Lo	10	Input 102 Hi	00
Packet CRC Hi		Input 102 Lo	02
Packet CRC Lo		Input 103 Hi	00
		Input 103 Lo	02
		Input 104 Hi	00
		Input 104 Lo	02
		Input 105 Hi	00
		Input 105 Lo	01
		Input 106 Hi	00
		Input 106 Lo	01
		Input 107 Hi	00
		Input 107 Lo	01
		Input 108 Hi	00
		Input 108 Lo	01
		Input 109 Hi	00
		Input 109 Lo	01
		Input 110 Hi	00
		Input 110 Lo	01
		Input 111 Hi	00
		Input 111 Lo	03
		Input 112 Hi	00
		Input 112 Lo	03
		Input 113 Hi	00
		Input 113 Lo	03
		Input 114 Hi	00
		Input 114 Lo	03
		Input 115 Hi	00
		Input 115 Lo	03
		Input 116 Hi	00
		Input 116 Lo	03
		Packet CRC Hi	
		Packet CRC Lo	

Table 11. Sample packet for read of MI Reg1 to Reg16.

Read MI Reg1 to Reg3

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	00	Byte Count	06
Register Address Lo	00	Input 101 Hi	00
Quantity of Input Reg. Hi	00	Input 101 Lo	01
Quantity of Input Reg. Lo	03	Input 102 Hi	00
Packet CRC Hi		Input 102 Lo	02
Packet CRC Lo		Input 103 Hi	00
		Input 103 Lo	02
		Packet CRC Hi	
		Packet CRC Lo	

Table 12. Sample packet for read of MI Reg1 to Reg3.

Table 12 shows the result of a request to read MI Reg1 to Reg3 (Function 4, MI 101–103 in map).

Read MI Reg15 to Reg18

Table 13 shows the result of a request to read MI Reg15 to Reg18 (Function 4, MI115–MI116 and MI201–MI202 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	00	Byte Count	08
Register Address Lo	0E	Input 115 Hi	00
Quantity of Input Reg. Hi	00	Input 115 Lo	01
Quantity of Input Reg. Lo	04	Input 116 Hi	00
Packet CRC Hi		Input 116 Lo	02
Packet CRC Lo		Input 201 Hi	00
		Input 201 Lo	01
		Input 202 Hi	00
		Input 202 Lo	03
		Packet CRC Hi	
		Packet CRC Lo	

Table 13. Sample packet for read of MI Reg15 to Reg18.

Write to MI Reg15

Table 14 shows the result of a write to MI Reg15 (Function 5 Single Write, MI115 in map). Note that the packet shows an error, since MI values are read only.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	05	Error Code	85
Register Address Hi	00	Exception Code	85
Register Address Lo	0E	Packet CRC Hi	
Quantity of Input Reg. Hi	00	Packet CRC Lo	
Quantity of Input Reg. Lo	01		
Packet CRC Hi			
Packet CRC Lo			

Table 14. Sample packet for write of MI Reg15.

Multi-State Input (MI) Enumeration

Each Multi-state Input has an associated Multi-state Input Configuration (MIC) object that defines how the raw analog input value is translated in an enumerated value. The controller database defines two of these MIC tables (Table 15 & 16)

	State #	State Name	Voltage Range
MIC1	1	Off	4-5
	2	On	0-2
	3	N/A	2-4

Table 15. Multi-state Input Configuration 1

	State #	State Name	Voltage Range
MIC2	1	Off	4-5
	2	On	0-2

Table 16. Multi-state Input Configuration 2

The MIC associated with each Multi-state Input is contained in the registers detailed in table 17.

Register	MI Input	Type	Default
40001-40003	MI-101 MIC	RW	MIC1
40004-40006	MI-102 MIC	RW	MIC1
40007-40009	MI-103 MIC	RW	MIC1
40010-40012	MI-104 MIC	RW	MIC1
40013-40015	MI-105 MIC	RW	MIC1
40016-40018	MI-106 MIC	RW	MIC1
40019-40021	MI-107 MIC	RW	MIC1
40022-40024	MI-108 MIC	RW	MIC1
40025-40027	MI-109 MIC	RW	MIC1
40028-40030	MI-110 MIC	RW	MIC1
40031-40033	MI-111 MIC	RW	MIC1
40034-40036	MI-112 MIC	RW	MIC1
40037-40039	MI-113 MIC	RW	MIC1
40040-40042	MI-114 MIC	RW	MIC1
40043-40045	MI-115 MIC	RW	MIC1
40046-40048	MI-116 MIC	RW	MIC1
40101-40103	MI-201 MIC	RW	MIC1
40204-40106	MI-202 MIC	RW	MIC1
40107-40109	MI-203 MIC	RW	MIC1
40110-40112	MI-204 MIC	RW	MIC1
40113-40115	MI-205 MIC	RW	MIC1
40116-40118	MI-206 MIC	RW	MIC1
40119-40121	MI-207 MIC	RW	MIC1
40122-40124	MI-208 MIC	RW	MIC1
40125-40127	MI-209 MIC	RW	MIC1
40128-40130	MI-210 MIC	RW	MIC1
40131-40133	MI-211 MIC	RW	MIC1
40134-40136	MI-212 MIC	RW	MIC1
40137-40139	MI-213 MIC	RW	MIC1
40140-40142	MI-214 MIC	RW	MIC1
40143-40145	MI-215 MIC	RW	MIC1
40146-40148	MI-216 MIC	RW	MIC1
40201-40203	MI-301 MIC	RW	MIC1
40204-40206	MI-302 MIC	RW	MIC1
40207-40209	MI-303 MIC	RW	MIC1
40210-40212	MI-304 MIC	RW	MIC1
40213-40215	MI-305 MIC	RW	MIC1
40216-40218	MI-306 MIC	RW	MIC1
40219-40221	MI-307 MIC	RW	MIC1
40222-40224	MI-308 MIC	RW	MIC1
40225-40227	MI-309 MIC	RW	MIC1
40228-40230	MI-310 MIC	RW	MIC1
40231-40233	MI-311 MIC	RW	MIC1
40234-40236	MI-312 MIC	RW	MIC1
40237-40239	MI-313 MIC	RW	MIC1
40240-40242	MI-314 MIC	RW	MIC1
40243-40245	MI-315 MIC	RW	MIC1
40246-40248	MI-316 MIC	RW	MIC1
40301-40303	MI-401 MIC	RW	MIC1
40304-40306	MI-402 MIC	RW	MIC1
40307-40309	MI-403 MIC	RW	MIC1
40310-40312	MI-404 MIC	RW	MIC1
40313-40315	MI-405 MIC	RW	MIC1
40316-40318	MI-406 MIC	RW	MIC1
40319-40321	MI-407 MIC	RW	MIC1
40322-40324	MI-408 MIC	RW	MIC1
40325-40327	MI-409 MIC	RW	MIC1
40328-40330	MI-410 MIC	RW	MIC1
40331-40333	MI-411 MIC	RW	MIC1
40334-40336	MI-412 MIC	RW	MIC1
40337-40339	MI-413 MIC	RW	MIC1
40340-40342	MI-414 MIC	RW	MIC1
40343-40345	MI-415 MIC	RW	MIC1
40346-40348	MI-416 MIC	RW	MIC1
40401-40403	MI-501 MIC	RW	MIC1
40404-40406	MI-502 MIC	RW	MIC1
40407-40409	Reserved	-	-
40410-40412	Reserved	-	-
40413-40415	Reserved	-	-
40416-40418	Reserved	-	-
40419-40421	Reserved	-	-
40422-40424	Reserved	-	-
40425-40427	Reserved	-	-
40428-40430	Reserved	-	-
40431-40433	Reserved	-	-
40434-40436	Reserved	-	-
40437-40439	Reserved	-	-
40440-40442	Reserved	-	-
40443-40445	Reserved	-	-
40446-40448	Reserved	-	-

Table 17. MIC associated with each Multi-state Input

The format of the Multi-state Input's associated MIC reference is shown in table 18. Changing the MIC reference changes how the Multi-state Input's value is interpreted. (Ex for MI-301).

Offset	Modbus Reg	DB Size	Property	Format	Range	Type
0	40201	WORD	Value From	Int	0 – 2 0 = MIC Standard 1 = reserved 2 = MIC Stepping	RW
			Configuration Ref.			
1	40202	WORD	Object Type	Word	183 [MIC]	RW
2	40203	WORD	Object Instance	Word	1 = MIC1 (GE Switch) 2 = MIC2 (On/Off)	RW
			3 registers per MI			

Table 18. Format of Multi-state Input associated MIC reference

Note: Configuration Reference registers (i.e. 40202 and 40203) must be written in a single write using Function 16.

Analog Input Registers

Lighting controller contains three on-board Analog Inputs and upto 16 remote Inputs (8 per Daylight Optimisation module . The present value of the Analog Input objects are mapped as indicated in table 19.

Analog Input (AI) Value- (Registers 101-148)

Register	AI Input	Type	Format
101 – 102	AI1 On-board	R	Float
103 – 104	AI2 On-board	R	Float
105 – 106	AI3 On-board	R	Float
117 – 118	AI-601	R	Float
119 – 120	AI-602	R	Float
121 – 122	AI-603	R	Float
123 – 124	AI-604	R	Float
125 – 126	AI-605	R	Float
127 – 128	AI-606	R	Float
129 – 130	AI-607	R	Float
131 – 132	AI-608	R	Float
133 – 134	AI-701	R	Float
135 – 136	AI-702	R	Float
137 – 138	AI-703	R	Float
139 – 140	AI-704	R	Float
141 – 142	AI-705	R	Float
143 – 144	AI-706	R	Float
145 – 146	AI-707	R	Float
147 – 148	AI-708	R	Float

Table 19. Analog Input Registers

Analog Input (AI) Object (Registers 1001 – 1272)

Table 20 shows different registers associated with each AI.

Register	AI Input	Type	Conf Ref
1001 – 1009	AI1 On-board	RW	
1010 – 1018	AI2 On-board	RW	
1019 – 1027	AI3 On-board	RW	
1101 – 1109	AI-601	RW	AIC1
1110 – 1118	AI-602	RW	AIC1
1119 – 1127	AI-603	RW	AIC1
1128 – 1136	AI-604	RW	AIC1
1137 – 1145	AI-605	RW	AIC1
1146 – 1154	AI-606	RW	AIC1
1155 – 1163	AI-607	RW	AIC1
1164 – 1172	AI-608	RW	AIC1
1201 – 1209	AI-701	RW	AIC1
1210 – 1218	AI-702	RW	AIC1
1219 – 1227	AI-703	RW	AIC1
1228 – 1236	AI-704	RW	AIC1
1237 – 1245	AI-705	RW	AIC1
1246 – 1254	AI-706	RW	AIC1
1255 – 1263	AI-707	RW	AIC1
1264 – 1272	AI-708	RW	AIC1

Table 20. Analog Input Property Registers

Analog Input (AI) Properties:

Each AI object exposes the following properties via Function codes 3/6/16. There are 6 properties available: Value, Calibration and Configuration Ref properties take multiple registers and Value & Reliability can only be written if OutOfService is True (OutofService means breaker is in manual mode). To remove a Configuration Ref, write 0 into the corresponding registers for object type and instance.

Table 21 shows different registers of AI1 object.

Offset	Modbus Regs	DB size	Property	Format	Range	Type
0	1001	Bit	OutOfService	Int	0 – 1	RW
1	1002	FLOAT Hi	Value *		0 – 65535	RW
2	1003	FLOAT Lo			0 – 65535	RW
3	1004	FLOAT Hi	Calibration		0 – 65535	RW
4	1005	FLOAT Lo			0 – 65535	RW
5	1006	Bit	Commissioned	Int	0 – 1	RW
6	1007	WORD	Reliability	Int	Enum	R
			Configuration Ref.			
7	1008	WORD	Object Type		176 [AIC]	RW
8	1009	WORD	Object Instance		1 – 65535 1 = Photodiode Linear 2 = Photoconductive 3 = Photodiode Linear (Inverse) 4 = Photoconductive (Inverse)	RW
			9 registers per AI			

Table 21. Property of AI1 object

Note:

- AIC reference registers 1008 and 1009 must be written in a single write using Function 16.
- Analog Input Configuration (AIC) object type in Hexadecimal is <00B0>

Sample packets

Table 22 indicates the result of read request for AI1 object

Request		**Response**	
Field Name	(Hex)	Field Name	(Hex)
Device Address	01	Device Address	01
Function	03	Function	03
Starting Address Hi	03	Byte Count	0E
Starting Address Lo	E8	Reg. Value Hi (1001)	00
Quantity of Input Reg. Hi	00	Reg. Value Lo (1001)	00
Quantity of Input Reg. Lo	07	Reg. Value Hi (1002)	41
Packet CRC Hi		Reg. Value Lo (1002)	A0
Packet CRC Lo		Reg. Value Hi (1003)	00
		Reg. Value Lo (1003)	00
		Reg. Value Hi (1004)	40
		Reg. Value Lo (1004)	A0
		Reg. Value Hi (1005)	00
		Reg. Value Lo (1005)	00
		Reg. Value Hi (1006)	00
		Reg. Value Lo (1006)	01
		Reg. Value Hi (1007)	00
		Reg. Value Lo (1007)	00
		Reg. Value Hi (1008)	00
		Reg. Value Low(1008)	0B
		Reg. Value Hi (1009)	00
		Reg. Value Low(1009)	01
		Packet CRC Hi	
		Packet CRC Lo	

Table 22 Sample packet for read of AI1 register

NOTE: Response for AI1 properties

- Reg1001 (OutOfService) 0x0000 – FALSE,
- Reg1002 – 1003 (Value) 0x41A00000 – 20.0
- Reg1004 – 1005 (Calibration) 0x40A00000 – 5.0,
- Reg1006 (Commission) 0x0001 – TRUE,
- Reg1007 (Reliability) 0x0000 – Normal,
- Reg1008 – 1009 (Configuration Reference) 0x000B0001 – AIC1

Table 23 shows the result of read request for AI601 object (Registers 1101-1109)

Read entire AI601 object – Registers 1101-1109

Request		**Response**	
Field Name	(Hex)	Field Name	(Hex)
Device Address	01	Device Address	01
Function	03	Function	03
Starting Address Hi	04	Byte Count	0E
Starting Address Lo	4C	Reg. Value Hi (1101)	00
Quantity of Input Reg. Hi	00	Reg. Value Lo (1101)	00
Quantity of Input Reg. Lo	07	Reg. Value Hi (1102)	42
Packet CRC Hi		Reg. Value Lo (1102)	C8
Packet CRC Lo		Reg. Value Hi (1103)	00
		Reg. Value Lo (1103)	00
		Reg. Value Hi (1104)	3F
		Reg. Value Lo (1104)	80
		Reg. Value Hi (1105)	00
		Reg. Value Lo (1105)	00
		Reg. Value Hi (1106)	00
		Reg. Value Lo (1106)	01
		Reg. Value Hi (1107)	00
		Reg. Value Lo (1107)	00
		Reg. Value Hi (1108)	00
		Reg. Value Lo (1108)	0B
		Reg. Value Hi (1109)	00
		Reg. Value Lo (1109)	01
		Packet CRC Hi	
		Packet CRC Lo	

Table 23 Sample packet for read of AI 601 register

NOTE: Response for AI601 properties

- Reg1001 (OutOfService) 0x0000 – FALSE
- Reg1002 – 1003 (Value) 0x42C80000 – 100.0
- Reg1004 – 1005 (Calibration) 0x3F800000 – 1.0
- Reg1006 (Commission) 0x0001 – TRUE
- Reg1007 (Reliability) 0x0000 – Normal
- Reg1008 – 1009 (Configuration Reference) 0x00B00001 – AIC1

Analog Variables (AV) – Physical Input Setup

The input value behavior of the two remote (Daylight Optimization) boards can be altered by adjusting the parameters represented by the Analog Variables AV601-AV632 & AV701-AV732.

Analog Variable (AV) Object – (Registers 1301-1364 & 1401-1464)

The Analog Input Parameters (i.e. AV Present Value) are mapped as follows (Table 24):

Input	Set Point (SP) Registers	Proportional Band Registers	Reset Rate (RR) Registers	Output Value (OP)* Register
AI-601	1301 – 1302 (AV601)	1317 – 1318 (AV609)	1333 – 1334 (AV617)	1349 – 1350 (AV625)
AI-602	1303 – 1304 (AV602)	1319 – 1320 (AV610)	1335 – 1336 (AV618)	1351 – 1352 (AV626)
AI-603	1305 – 1306 (AV603)	1321 – 1322 (AV611)	1337 – 1338 (AV619)	1353 – 1354 (AV627)
AI-604	1307 – 1308 (AV604)	1323 – 1324 (AV612)	1339 – 1340 (AV620)	1355 – 1356 (AV628)
AI-605	1309 – 1310 (AV605)	1325 – 1326 (AV613)	1341 – 1342 (AV621)	1357 – 1358 (AV629)
AI-606	1311 – 1312 (AV606)	1327 – 1328 (AV614)	1343 – 1344 (AV622)	1359 – 1360 (AV630)
AI-607	1313 – 1314 (AV607)	1329 – 1330 (AV615)	1345 – 1346 (AV623)	1361 – 1362 (AV631)
AI-608	1315 – 1316 (AV608)	1331 – 1332 (AV616)	1347 – 1348 (AV624)	1363 – 1364 (AV632)
AI-701	1401 – 1402 (AV701)	1417 – 1418 (AV709)	1433 – 1434 (AV717)	1449 – 1450 (AV725)
AI-702	1403 – 1404 (AV702)	1419 – 1420 (AV710)	1435 – 1436 (AV718)	1451 – 1452 (AV726)
AI-703	1405 – 1406 (AV703)	1421 – 1422 (AV711)	1437 – 1438 (AV719)	1453 – 1454 (AV727)
AI-704	1407 – 1408 (AV704)	1423 – 1424 (AV712)	1439 – 1440 (AV720)	1455 – 1456 (AV728)
AI-705	1409 – 1410 (AV705)	1425 – 1426 (AV713)	1441 – 1442 (AV721)	1457 – 1458 (AV729)
AI-706	1411 – 1412 (AV706)	1427 – 1428 (AV714)	1443 – 1444 (AV722)	1459 – 1460 (AV730)
AI-707	1413 – 1414 (AV707)	1429 – 1430 (AV715)	1445 – 1446 (AV723)	1461 – 1462 (AV731)
AI-708	1415 – 1416 (AV708)	1431 – 1432 (AV716)	1447 – 1448 (AV724)	1463 – 1464 (AV732)

Table 24 Analog Variable Registers

*Setting SP= -1 allows writing of the Output Value.
**Note: do NOT write to any of these registers without specific knowledge of the consequence

Analog Variable (AV) Properties

Each AV object exposes the following properties (Table25) via Function codes 3/6/16.

Offset	Modbus Regs	DB size	Property	Format	Range	Type
0	1301 – 1302	FLOAT Hi	Value		0 – 65535	RW
			2 registers per AV			

Table 25. Analog Variable Properties

Physical Output Registers (Coils)

Lighting controller supports up to 66 breaker outputs (BO). Actual value registers start at 1, commanded by Function Codes 01, 05, and 15 and are mapped to Binary Outputs (BO) 1–66. The value in the register represents the Binary Output's present value when read. Writes go to priority level 5. Writes will only be accepted if the 'OutOfService' flag is set for that particular output.

A value of 1 represents ON and a value of 0 represents OFF.

The physical output registers (coils) are listed in Table 26.

Reg.	Contents	Type	Format	BO Instance
1	BO1 Value	RW	Bit	01
2	BO2 Value	RW	Bit	02
3	BO3 Value	RW	Bit	03
4	BO4 Value	RW	Bit	04
5	BO5 Value	RW	Bit	05
6	BO6 Value	RW	Bit	06
7	BO7 Value	RW	Bit	07
8	BO8 Value	RW	Bit	08
9	BO9 Value	RW	Bit	09
10	BO10 Value	RW	Bit	10
11	BO11 Value	RW	Bit	11
12	BO12 Value	RW	Bit	12
13	BO13 Value	RW	Bit	13
14	BO14 Value	RW	Bit	14
15	BO15 Value	RW	Bit	15
16	BO16 Value	RW	Bit	16
17	BO17 Value	RW	Bit	17
18	BO18 Value	RW	Bit	18
19	BO19 Value	RW	Bit	19
20	BO20 Value	RW	Bit	20
21	BO21 Value	RW	Bit	21
22	BO22 Value	RW	Bit	22
23	BO23 Value	RW	Bit	23
24	BO24 Value	RW	Bit	24
25	BO25 Value	RW	Bit	25
26	BO26 Value	RW	Bit	26
27	BO27 Value	RW	Bit	27
28	BO28 Value	RW	Bit	28
29	BO29 Value	RW	Bit	29
30	BO30 Value	RW	Bit	30
31	BO31 Value	RW	Bit	31
32	BO32 Value	RW	Bit	32
33	BO33 Value	RW	Bit	33
34	BO34 Value	RW	Bit	34
35	BO35 Value	RW	Bit	35
36	BO36 Value	RW	Bit	36
37	BO37 Value	RW	Bit	37
38	BO38 Value	RW	Bit	38
39	BO39 Value	RW	Bit	39
40	BO40 Value	RW	Bit	40
41	BO41 Value	RW	Bit	41
42	BO42 Value	RW	Bit	42
43	BO43 Value	RW	Bit	43
44	BO44 Value	RW	Bit	44
45	BO45 Value	RW	Bit	45
46	BO46 Value	RW	Bit	46
47	BO47 Value	RW	Bit	47
48	BO48 Value	RW	Bit	48
49	BO49 Value	RW	Bit	49
50	BO50 Value	RW	Bit	50
51	BO51 Value	RW	Bit	51
52	BO52 Value	RW	Bit	52
53	BO53 Value	RW	Bit	53
54	BO54 Value	RW	Bit	54
55	BO55 Value	RW	Bit	55
56	BO56 Value	RW	Bit	56
57	BO57 Value	RW	Bit	57
58	BO58 Value	RW	Bit	58
59	BO59 Value	RW	Bit	59
60	BO60 Value	RW	Bit	60
61	BO61 Value	RW	Bit	61
62	BO62 Value	RW	Bit	62
63	BO63 Value	RW	Bit	63
64	BO64 Value	RW	Bit	64
65	BO65 Value	RW	Bit	65
66	BO66 Value	RW	Bit	66

Table 26. List of physical output registers (coils).

Sample Packets

The following are examples of packets generated by requests for specific groups of outputs.

Read Coil Status of BO Reg2 to Reg17

Table 27 shows the result of a request to read the coil status of BO Reg2–Reg17 (Function 1, BO2–17 in map). Note that the status of outputs 9–2 is shown as the byte value 0xAC or binary 1010 1100 and output 2 is the LSB.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	00	Byte Count	02
Register Address Lo	01	Output Status 9–2	AC
Quantity of Outputs Hi	00	Output Status 17–10	0A
Quantity of Outputs Lo	10	Packet CRC Hi	
Packet CRC Hi		Packet CRC Lo	
Packet CRC Lo			

Table 27. Sample packet for coil status of BO Reg2 to Reg17.

Read Coil Status of BO Reg2 to Reg19

Table 28 shows the result of a request to read the coil status of BO Reg2–Reg19 (Function 1, BO2–19 in map). Note that the status of outputs 19–18 is shown as the byte value 0x03 or binary 0000 0011 and output 18 is the LSB. The six remaining high-order bits are zero filled.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	00	Byte Count	03
Register Address Lo	01	Output Status 9–2	AC
Quantity of Outputs Hi	00	Output Status 17–10	0A
Quantity of Outputs Lo	12	Output Status 19–18	03
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 28. Sample packet for coil status of BO Reg2 to Reg19.

Read Coil Status of BO Reg62 to Reg66

Table 29 shows the result of a request to read the coil status of BO Reg62–Reg66 (Function 1, BO62–66). Note that the status of outputs 66–62 is shown as the byte value 0x37 or binary 0001 0111 and output 62 is the LSB. The remaining two high-order bits are zero filled.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	00	Byte Count	03
Register Address Lo	3D	Output Status 66–62	17
Quantity of Outputs Hi	00	Packet CRC Hi	
Quantity of Outputs Lo	05	Packet CRC Lo	
Packet CRC Hi			
Packet CRC Lo			

Table 29. Sample packet for coil status of BO Reg62 to Reg 66.

The output values are

Reg62: BO62 Value – ON

Reg63: BO63 Value – ON

Reg64: BO64 Value – ON

Reg65: BO65 Value – OFF

Reg66: BO66 Value – ON

Physical Output Registers (OutOfService)

Actual value registers start at 201, commanded by Function Code 01, 05, or 15 and mapped to Binary Outputs (BO) 1–66. The address space is reserved to 900. The value in the register represents the Binary Output's OutOfService value. A value of 1 corresponds to manual mode, 0 to auto mode.The physical output registers (OutOfService) are listed in Table 30.

Reg.	Contents	Type	Format	BO Instance
201	BO1 OutOfService	RW	ON/OFF	01
202	BO2 OutOfService	RW	ON/OFF	02
203	BO3 OutOfService	RW	ON/OFF	03
204	BO4 OutOfService	RW	ON/OFF	04
205	BO5 OutOfService	RW	ON/OFF	05
206	BO6 OutOfService	RW	ON/OFF	06
207	BO7 OutOfService	RW	ON/OFF	07
208	BO8 OutOfService	RW	ON/OFF	08
209	BO9 OutOfService	RW	ON/OFF	09
210	BO10 OutOfService	RW	ON/OFF	10
211	BO11 OutOfService	RW	ON/OFF	11
212	BO12 OutOfService	RW	ON/OFF	12
213	BO13 OutOfService	RW	ON/OFF	13
214	BO14 OutOfService	RW	ON/OFF	14
215	BO15 OutOfService	RW	ON/OFF	15
216	BO16 OutOfService	RW	ON/OFF	16
217	BO17 OutOfService	RW	ON/OFF	17
218	BO18 OutOfService	RW	ON/OFF	18
219	BO19 OutOfService	RW	ON/OFF	19
220	BO20 OutOfService	RW	ON/OFF	20
221	BO21 OutOfService	RW	ON/OFF	21
222	BO22 OutOfService	RW	ON/OFF	22
223	BO23 OutOfService	RW	ON/OFF	23
224	BO24 OutOfService	RW	ON/OFF	24
225	BO25 OutOfService	RW	ON/OFF	25
226	BO26 OutOfService	RW	ON/OFF	26
227	BO27 OutOfService	RW	ON/OFF	27
228	BO28 OutOfService	RW	ON/OFF	28
229	BO29 OutOfService	RW	ON/OFF	29
230	BO30 OutOfService	RW	ON/OFF	30
231	BO31 OutOfService	RW	ON/OFF	31
232	BO32 OutOfService	RW	ON/OFF	32
233	BO33 OutOfService	RW	ON/OFF	33
234	BO34 OutOfService	RW	ON/OFF	34
235	BO35 OutOfService	RW	ON/OFF	35
236	BO36 OutOfService	RW	ON/OFF	36
237	BO37 OutOfService	RW	ON/OFF	37
238	BO38 OutOfService	RW	ON/OFF	38
239	BO39 OutOfService	RW	ON/OFF	39
240	BO40 OutOfService	RW	ON/OFF	40
241	BO41 OutOfService	RW	ON/OFF	41
242	BO42 OutOfService	RW	ON/OFF	42
243	BO43 OutOfService	RW	ON/OFF	43
244	BO44 OutOfService	RW	ON/OFF	44
245	BO45 OutOfService	RW	ON/OFF	45
246	BO46 OutOfService	RW	ON/OFF	46
247	BO47 OutOfService	RW	ON/OFF	47
248	BO48 OutOfService	RW	ON/OFF	48
249	BO49 OutOfService	RW	ON/OFF	49
250	BO50 OutOfService	RW	ON/OFF	50
251	BO51 OutOfService	RW	ON/OFF	51

Reg.	Contents	Type	Format	BO Instance
252	BO52 OutOfService	RW	ON/OFF	52
253	BO53 OutOfService	RW	ON/OFF	53
254	BO54 OutOfService	RW	ON/OFF	54
255	BO55 OutOfService	RW	ON/OFF	55
256	BO56 OutOfService	RW	ON/OFF	56
257	BO57 OutOfService	RW	ON/OFF	57
258	BO58 OutOfService	RW	ON/OFF	58
259	BO59 OutOfService	RW	ON/OFF	59
260	BO60 OutOfService	RW	ON/OFF	60
261	BO61 OutOfService	RW	ON/OFF	61
262	BO62 OutOfService	RW	ON/OFF	62
263	BO63 OutOfService	RW	ON/OFF	63
264	BO64 OutOfService	RW	ON/OFF	64
265	BO65 OutOfService	RW	ON/OFF	65
266	BO66 OutOfService	RW	ON/OFF	66

Table 30. List of physical output registers (OutOfService).

Sample Packets

The following are examples of requests to specific groups of output registers.

Write to Output Reg201

Table 31 shows the result of a write request to output OutOfService Reg201 (Function 5 Single Write, BO1 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	0F	Function	0F
Register Address Hi	DB	Register Address Hi	DB
Register Address Lo	00	Register Address Lo	00
Quantity of Outputs Hi	00	Quantity of Outputs Hi	00
Quantity of Outputs Lo	0A	Quantity of Outputs Lo	0A
Byte Count	02	Packet CRC Hi	
Output Value Hi	CD	Packet CRC Lo	
Output Value Lo	03		
Packet CRC Hi			
Packet CRC Lo			

Table 31. Sample packet for multiple write to output Reg220–Reg229.

Multiple Write Output Reg220–Reg229

Table 32 shows the result of a multiple write to BO OutOfService Reg220–229 (Function 15 Multiple Write, BO 20–29 in map).

Note that the Hi byte 0xCD addresses outputs 27–20, with the least-significant bit addressing the lowest output 20 and the most-significant bit addressing the highest output 27. The Lo byte 03 addresses outputs 29–28, with the least-significant bit addressing the lowest output 28. Unused bits in the last data byte are padded with zeros

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	05	Function	05
Register Address Hi	00	Register Address Hi	00
Register Address Lo	C8	Register Address Lo	C8
Output Value Hi	FF	Output Value Hi	FF
Output Value Lo	00	Output Value Lo	00
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 32. Sample packet for write to output Reg201.

Binary Output (BO) Object – (Registers 2001 – 3260)

Each Binary Output object contains 12 properties that are exposed to MODBUS. These object properties are writable via Function Codes 6 & 16 and readable by Function Code 3. Writes to the BO Value occur at BACnet Priority Level 5 (Critical Equipment Control). Table 33 shows list of registers associated with B01-B066.

Register	BO Object	Type
2001 – 2018	BO1	RW
2019 – 2036	BO2	RW
2037 – 2054	BO3	RW
2055 – 2072	BO4	RW
2073 – 2090	BO5	RW
2091 – 2108	BO6	RW
2109 – 2126	BO7	RW
2127 – 2144	BO8	RW
2145 – 2162	BO9	RW
2163 – 2180	BO10	RW
2181 – 2198	BO11	RW
2199 – 2216	BO12	RW
2217 – 2234	BO13	RW
2235 – 2252	BO14	RW
2253 – 2270	BO15	RW
2271 – 2288	BO16	RW
2289 – 2306	BO17	RW
2307 – 2324	BO18	RW
2325 – 2342	BO19	RW
2343 – 2360	BO20	RW
2361 – 2378	BO21	RW
2379 – 2396	BO22	RW
2397 – 2414	BO23	RW
2415 – 2432	BO24	RW
2433 – 2450	BO25	RW
2451 – 2468	BO26	RW
2469 – 2486	BO27	RW
2487 – 2504	BO28	RW
2505 – 2522	BO29	RW
2523 – 2540	BO30	RW
2541 – 2558	BO31	RW

2559 – 2576	BO32	RW
2577 – 2594	BO33	RW
2595 – 2612	BO34	RW
2613 – 2630	BO35	RW
2631 – 2648	BO36	RW
2649 – 2666	BO37	RW
2667 – 2684	BO38	RW
2685 – 2702	BO39	RW
2703 – 2720	BO40	RW
2721 – 2738	BO41	RW
2739 – 2756	BO42	RW
2757 – 2774	BO43	RW
2775 – 2792	BO44	RW
2793 – 2810	BO45	RW
2811 – 2828	BO46	RW
2829 – 2846	BO47	RW
2847 – 2864	BO48	RW
2865 – 2882	BO49	RW
2883 – 2900	BO50	RW
2901 – 2918	BO51	RW
2919 – 2936	BO52	RW
2937 – 2954	BO53	RW
2955 – 2972	BO54	RW
2973 – 2990	BO55	RW
2991 – 3008	BO56	RW
3009 – 3026	BO57	RW
3027 – 3044	BO58	RW
3045 – 3062	BO59	RW
3063 – 3080	BO60	RW
3081 – 3098	BO61	RW
3099 – 3116	BO62	RW
3117 – 3134	BO63	RW
3135 – 3152	BO64	RW
3153 – 3170	BO65	RW
3171 – 3188	BO66	RW

Table 33. Binary Output (BO) Registers

Offset	Modbus regs	DB size	Property	Format	Range	Type
0	2001	Bit	OutOfService	Int	0–1	RW
1	2002	Byte	Value	Int	0–1	RW
2	2003	Bit	Commissioned	Int	0–1	RW
3	2004	Word	Reliability	Int	Enum	R
4	2005–2006	Dword	MinimumOffTime	Word	0–65536	RW
6	2007–2008	Dword	MinimumOnTime	Word	0–65536	RW
8	2009	Byte	DefaultValue	Word	0–65536	RW
9	2010–2011	Dword	StartDelayTime	Word	0–65536	RW
11	2012	Byte	FbackValue	Word	0–65536	RW
12	2013	Byte	FlickWarnTime	Word	0–65536	RW
13	2014	Word	FlickWarnPAEnable	Int	On/Off	RW
14	2015	Word	OverrideTime	Word	0–65536	RW

Table 34. Address space of binary output registers.

Binary Output (BO) Properties:

Each Binary Output (BO) object exposes the following object properties (Table 34).

Sample Packets

Following are examples of function requests to binary output registers.

Read All BO1 Properties

Table.35 is an example of a read of all BO1 properties: OutOfService, Value, Commissioned, Reliability and MinimumOffTime Reg2001–Reg2006 (Function 3, BO1in map).

Request		Response	
Field Name		Hex	
Device Address	01	Field Name	Hex
Function	03	Device Address	01
Register Address Hi	00	Function	03
Register Address Lo	00	Byte Count	0C
Registers Hi	00	Register Value Hi (2001)	00
Registers Lo	06	Register Value Lo (2001)	00
Packet CRC Hi		Register Value Hi (2002)	00
Packet CRC Lo		Register Value Lo (2002)	01
		Register Value Hi (2003)	00
		Register Value Lo (2003)	00
		Register Value Hi (2004)	00
		Register Value Lo (2004)	00
		Register Value Hi (2005)	00
		Register Value Lo (2005)	00
		Register Value Hi (2006)	00
		Register Value Lo (2006)	3C
		Packet CRC Hi	

Table 35. Sample packets for read of all BO1 properties.

The response is as follows:

Reg2001 (OutOfService): 0x0000, FALSE,

Reg2002 (Value): 0x0001, ON

Reg2003 (Commissioned): 0x000, FALSE

Reg2004 (Reliability): 0x0000, Normal,

Reg2005 (MinimumOffTime): 0x0003C, 60sec

Single Write to BO1 Override Time

Table.36 is an example of a single write of 5 minutes (0x0005) to BO1 Override Time Reg.2015 (Function 6, BO1 in map).

Request		Response	
Field Name	Hex	Field Name	Hex
Device Address	01	Device Address	01
Function	06	Function	06
Register Address Hi	07	Register Address Hi	07
Register Address Lo	DE	Register Address Lo	DE
Registers Hi	00	Registers Hi	00
Registers Lo	05	Registers Lo	05
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 36. Sample packets for single write to BO1 overwrite time.

Multiple Write to Binary Output

Table.37 is an example of a multiple write to BO OutOfService and Value 1–3 (Function 16, BO 1in map). This example writes OutOfService to TRUE and Value to 0x0000 (OFF).

Request		Response	
Field Name	Hex	Field Name	Hex
Device Address	01	Device Address	01
Function	10	Function	10
Register Address Hi	07	Register Address Hi	00
Register Address Lo	D0	Register Address Lo	00
Registers Hi	00	Registers Hi	00
Registers Lo	02	Registers Lo	02
Register Value 1 Hi	00	Packet CRC Hi	
Register Value 1 Lo	01	Packet CRC Lo	
Register Value 2 Hi	00		
Register Value 2 Lo	00		
Packet CRC Hi			
Packet CRC Lo			

Table 37. Sample packets for multiple write to binary output.

Feedback Registers

The feedback value of the Binary outputs(BO) is mapped starting at Modbus register 1 and are read by function code 02. Actual value (Feedback) registers start at 1, are read by Function Code 02, and are mapped to Binary Outputs 1–66. Note that the value of each register is the combined value of the current feedback input value and the current command of that output. A bit value of 0 represents the feedback OFF and a 1 represents the Feedback ON.

The feedback registers are listed in Table 38.

Reg.	Contents	Type	Format	BO Instance
1	BO1 Feedback	R	Bit	01
2	BO2 Feedback	R	Bit	02
3	BO3 Feedback	R	Bit	03
4	BO4 Feedback	R	Bit	04
5	BO5 Feedback	R	Bit	05
6	BO6 Feedback	R	Bit	06
7	BO7 Feedback	R	Bit	07
8	BO8 Feedback	R	Bit	08
9	BO9 Feedback	R	Bit	09
10	BO10 Feedback	R	Bit	10
11	BO11 Feedback	R	Bit	11
12	BO12 Feedback	R	Bit	12
13	BO13 Feedback	R	Bit	13
14	BO14 Feedback	R	Bit	14
15	BO15 Feedback	R	Bit	15
16	BO16 Feedback	R	Bit	16
17	BO17 Feedback	R	Bit	17
18	BO18 Feedback	R	Bit	18
19	BO19 Feedback	R	Bit	19
20	BO20 Feedback	R	Bit	20
21	BO21 Feedback	R	Bit	21
22	BO22 Feedback	R	Bit	22
23	BO23 Feedback	R	Bit	23
24	BO24 Feedback	R	Bit	24
25	BO25 Feedback	R	Bit	25
26	BO26 Feedback	R	Bit	26
27	BO27 Feedback	R	Bit	27
28	BO28 Feedback	R	Bit	28
29	BO29 Feedback	R	Bit	29
30	BO30 Feedback	R	Bit	30
31	BO31 Feedback	R	Bit	31
32	BO32 Feedback	R	Bit	32
33	BO33 Feedback	R	Bit	33
34	BO34 Feedback	R	Bit	34
35	BO35 Feedback	R	Bit	35
36	BO36 Feedback	R	Bit	36
37	BO37 Feedback	R	Bit	37
38	BO38 Feedback	R	Bit	38
39	BO39 Feedback	R	Bit	39
40	BO40 Feedback	R	Bit	40
41	BO41 Feedback	R	Bit	41
42	BO42 Feedback	R	Bit	42
43	BO43 Feedback	R	Bit	43
44	BO44 Feedback	R	Bit	44
45	BO45 Feedback	R	Bit	45
46	BO46 Feedback	R	Bit	46
47	BO47 Feedback	R	Bit	47
48	BO48 Feedback	R	Bit	48
49	BO49 Feedback	R	Bit	49
50	BO50 Feedback	R	Bit	50
51	BO51 Feedback	R	Bit	51
52	BO52 Feedback	R	Bit	52
53	BO53 Feedback	R	Bit	53
54	BO54 Feedback	R	Bit	54
55	BO55 Feedback	R	Bit	55
56	BO56 Feedback	R	Bit	56
57	BO57 Feedback	R	Bit	57
58	BO58 Feedback	R	Bit	58
59	BO59 Feedback	R	Bit	59
60	BO60 Feedback	R	Bit	60
61	BO61 Feedback	R	Bit	61
62	BO62 Feedback	R	Bit	62
63	BO63 Feedback	R	Bit	63
64	BO64 Feedback	R	Bit	64
65	BO65 Feedback	R	Bit	65
66	BO66 Feedback	R	Bit	66

Table 38. List of feedback registers.

Sample Packets

Table 39 is an example of a request to read for output Feedback instances 1–19 (Function 2, BO 1–19 in map).

Note that the feedback status of inputs 8–1 is shown as the byte value 0xAC or binary 1010 1100. Input 8 is the MSB of this byte and input 1 is the LSB. The feedback status input 19–17 is shown as the byte value 0x06 or binary 0000 0110. Input 19 is the third bit position from the left and input 17 is the LSB.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	02	Function	0F
Register Address Hi	00	Byte Count	03
Register Address Lo	00	Input Status 8–1	AC
Quantity of Inputs Hi	00	Input Status 16–9	DB
Quantity of Inputs Lo	0A	Input Status 19–17	06
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 39. Sample packets for output feedback instances 1–19

Special Registers

In addition to the 66 breaker outputs, the lighting controller has several special outputs (Table 40) to control other features on the main board.

These special outputs are mapped to MODBUS coil registers 67-70 and are accessed by function code 01/05/15.

Note: These outputs are not for general use. Don't write to these registers without detailed knowledge of their purpose, operation and possible consequences.

Reg	Contents	Units / Range	Type
67	[BO67] – Output Sweep	On/Off	R/W
68	[BO68] - Device Alarm Status	On/Off	R/W
69	[BO69] - Voltage Input	On/Off	R/W
70	[BO70] - MSP430 Reset	On/Off	R/W

Table 40.Special output Registers

Sample packets

The following are samples of packets generated by requests for specific groups of outputs.

Read BO Registers 67 to 70

Table 41 is an example of a request to read BO67-BO70 (Function 1, BO67 – O70).

Request		Response	
Field Name	Hex	Field Name	Hex
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	00	Byte Count	01
Register Address Lo	42	Outputs Status 70 – 67	0E
Quantity of Outputs Hi	00	Packet CRC Hi	
Quantity of Outputs Lo	04	Packet CRC Lo	
Packet CRC Hi			
Packet CRC Lo			

Table 41. Request to read BO67-BO70

Write "On" to output Value Reg67 (Function 5 Single Write, BO67 in map). Table 42 is an example of a single write to BO67.

Request		Response	
Field Name	Hex	Field Name	Hex
Device Address	01	Device Address	01
Function	05	Function	05
Register Address Hi	00	Register Address Hi	00
Register Address Lo	42	Register Address Lo	42
Output Value Hi	FF	Output Value Hi	FF
Output Value Lo	00	Output Value Lo	00
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 42. Request to write Register 67

Binary Variable (BV) Object

The GE Default Database contains 66 Binary Variable objects that may be used as a Light Switch (Input Override) for each Lighting Output (i.e. Breaker) in an LG (Lighting Group).

Reg	Contents	Type	Format		Reg	Contents	Type	Format
3401	BV1 Value	RW	On/Off		3434	BV34 Value	RW	On/Off
3402	BV2 Value	RW	On/Off		3435	BV35 Value	RW	On/Off
3403	BV3 Value	RW	On/Off		3436	BV36 Value	RW	On/Off
3404	BV4 Value	RW	On/Off		3437	BV37 Value	RW	On/Off
3405	BV5 Value	RW	On/Off		3438	BV38 Value	RW	On/Off
3406	BV6 Value	RW	On/Off		3439	BV39 Value	RW	On/Off
3407	BV7 Value	RW	On/Off		3440	BV40 Value	RW	On/Off
3408	BV8 Value	RW	On/Off		3441	BV41 Value	RW	On/Off
3409	BV9 Value	RW	On/Off		3442	BV42 Value	RW	On/Off
3410	BV10 Value	RW	On/Off		3443	BV43 Value	RW	On/Off
3411	BV11 Value	RW	On/Off		3444	BV44 Value	RW	On/Off
3412	BV12 Value	RW	On/Off		3445	BV45 Value	RW	On/Off
3413	BV13 Value	RW	On/Off		3446	BV46 Value	RW	On/Off
3414	BV14 Value	RW	On/Off		3447	BV47 Value	RW	On/Off
3415	BV15 Value	RW	On/Off		3448	BV48 Value	RW	On/Off
3416	BV16 Value	RW	On/Off		3449	BV49 Value	RW	On/Off
3417	BV17 Value	RW	On/Off		3450	BV50 Value	RW	On/Off
3418	BV18 Value	RW	On/Off		3451	BV51 Value	RW	On/Off
3419	BV19 Value	RW	On/Off		3452	BV52 Value	RW	On/Off
3420	BV20 Value	RW	On/Off		3453	BV53 Value	RW	On/Off
3421	BV21 Value	RW	On/Off		3454	BV54 Value	RW	On/Off
3422	BV22 Value	RW	On/Off		3455	BV55 Value	RW	On/Off

			f					f
3423	BV23 Value	RW	On/Off		3456	BV56 Value	RW	On/Off
3424	BV24 Value	RW	On/Off		3457	BV57 Value	RW	On/Off
3425	BV25 Value	RW	On/Off		3458	BV58 Value	RW	On/Off
3426	BV26 Value	RW	On/Off		3459	BV59 Value	RW	On/Off
3427	BV27 Value	RW	On/Off		3460	BV60 Value	RW	On/Off
3428	BV28 Value	RW	On/Off		3461	BV61 Value	RW	On/Off
3429	BV29 Value	RW	On/Off		3462	BV62 Value	RW	On/Off
3430	BV30 Value	RW	On/Off		3463	BV63 Value	RW	On/Off
3431	BV31 Value	RW	On/Off		3464	BV64 Value	RW	On/Off
3432	BV32 Value	RW	On/Off		3465	BV65 Value	RW	On/Off
3433	BV33 Value	RW	On/Off		3466	BV66 Value	RW	On/Off

Binary Variable (BV) Properties

Each Binary Variable (BV) Object exposes the following object properties:

Offset	Modbus regs	DB size	Property	Format	Range	Type
0	3401	BYTE	Value *	Int	0 - 1	RW
			1 register per BV			

Lighting Group Objects

Individual binary outputs can be grouped together to represent the logical layout of a building installation. Sixteen lighting groups (LG) are available and each one can contain a group of 33 outputs.

Lighting Group Output Registers (Coils)

Actual value registers start at 301. The Lighting Group and OutOfService property can be commanded by Function Code 05 and 15 and read by Function Code 1 and are mapped to Lighting Group Instances 1–16. The value in the register represents the Lighting Group's PresentValue.The Lighting Group output registers are listed inTable 43.

Reg.	Contents	Type	Format	Instance
301	LG01 Value	RW	On/Off	01
302	LG02 Value	RW	On/Off	02
303	LG03 Value	RW	On/Off	03
304	LG04 Value	RW	On/Off	04
305	LG05 Value	RW	On/Off	05
306	LG06 Value	RW	On/Off	06
307	LG07 Value	RW	On/Off	07
308	LG08 Value	RW	On/Off	08
309	LG09 Value	RW	On/Off	09
310	LG10 Value	RW	On/Off	10
311	LG11 Value	RW	On/Off	11
312	LG12 Value	RW	On/Off	12
313	LG13 Value	RW	On/Off	13
314	LG14 Value	RW	On/Off	14
315	LG15 Value	RW	On/Off	15
316	LG16 Value	RW	On/Off	16

Table 43. Lighting Group output registers.

Sample Packets

Table44 is an example of a request for LG Value Reg302 to Reg305 (Function Code 1, LG2 – LG5 in map).

Note that the status of outputs 5–2 is hex 0B or binary 0000 1011, where LG302 is the LSB and LG305 is the fourth bit from the left. Unused bits are padded as zero.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	01	Byte Count	01
Register Address Lo	2D	Output Status 5–2	0B
Quantity of LG Hi	00	Packet CRC Hi	
Quantity of LG Lo	04	Packet CRC Lo	
Packet CRC Hi			
Packet CRC Lo			

Table 44. Sample packets for LG Value Reg302 to Reg305.

Lighting Group Registers (OutOfService)

The Lighting Group OutOfService registers are listed in Table45.

Reg.	Contents	Type	Format	Instance
401	LG01 OutOfService	RW	On/Off	01
402	LG02 OutOfService	RW	On/Off	02
403	LG03 OutOfService	RW	On/Off	03
404	LG04 OutOfService	RW	On/Off	04
405	LG05 OutOfService	RW	On/Off	05
406	LG06 OutOfService	RW	On/Off	06
407	LG07 OutOfService	RW	On/Off	07
408	LG08 OutOfService	RW	On/Off	08
409	LG09 OutOfService	RW	On/Off	09
410	LG10 OutOfService	RW	On/Off	10
411	LG11 OutOfService	RW	On/Off	11
412	LG12 OutOfService	RW	On/Off	12
413	LG13 OutOfService	RW	On/Off	13
414	LG14 OutOfService	RW	On/Off	14
415	LG15 OutOfService	RW	On/Off	15
416	LG16 OutOfService	RW	On/Off	16

Table 45. Lighting Group OutOfService registers.

Sample Packets

Table 46 is an example of a request for LG OutOfService status instance 401–416 (Function Code 1, LG1–16 in map).

Note that the LG OutOfService Status of 408–401 is shown as hex AB or binary 1010 1011, where LG 401 is the LSB and LG408 is the MSB.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	01	Byte Count	02
Register Address Lo	90	OutOfService Reg408–401	AB
Quantity of LG Hi	00	OutOfService Reg416–409	AA
Quantity of LG Lo	10	Packet CRC Hi	
Packet CRC Hi		Packet CRC Lo	
Packet CRC Lo			

Table 46. Sample packets for LG out of Service instance 401–416.

Lighting Group Setup Registers

The Lighting Group setup parameters are assigned a block of holding registers starting at 4001 as indicated in table 47. Lighting Group properties can be commanded

by Function Code 06 and 16 and read by Function 3. Write to LG value is at priority level 1 by default.

Note that any reference in this object that does not exist is shown as 0xFFFF in all three registers. To delete a reference, write 0x0000 to all three registers. The same applies to deleting an entry in any of the two arrays in this object. Lighting Outputs and Common Group entries are sorted in object execution time. Therefore, multiple entries can only be deleted in one write.

The template shown in table 48 is repeated 16 times in this assigned space.

Register	LG Object	Type		Register	BO Object	Type
4001 – 4363	LG1	RW		6905 – 7267	LG9	RW
4364 – 4726	LG2	RW		7268 – 7630	LG10	RW
4727 – 5089	LG3	RW		7631 – 7993	LG11	RW
5090 – 5452	LG4	RW		7994 – 8356	LG12	RW
5453 – 5815	LG5	RW		8357 – 8719	LG13	RW
5816 – 6179	LG6	RW		8720 – 9082	LG14	RW
6179 – 6542	LG7	RW		9083 – 9445	LG15	RW
6542 – 6904	LG8	RW		9446 – 9808	LG16	RW

Table 47 different lighting group registers

Offset	Modbus #regs	DB size	Property	Format	Range	Type
0	4001	BYTE	Out of Service	Int	0 – 1	RW
1	4002	BYTE	Value *	Int	0 – 1	RW
2	4003	BYTE	Current Priority	Int	Enum (1 – 6)	R
3	4004	WORD	Reliability	Int	Enum	R
			Override Reference			
4	4005		Panel Number	Int	0 – 99	RW
5	4006		Object Type	Int	0 – 65535 BI= 3 BV= 5	RW
6	4007		Object Instance	Word	1 – 65535	RW
7	4008	WORD	Override Time	WORD	0 – 65535	RW
			Photocell Reference			
8	4009		Panel Number	Word	0 – 99	RW
9	4010		Object Type	Int	0 – 65535 AI= 0 AV= 2 BI= 3 MI= 13	RW
10	4011		Object Instance	Word	1 – 65535	RW
11	4012 – 4013		Photocell On Val	WORD	0 – 65535	RW
13	4014 – 4015		Photocell Off Val	WORD	0 – 65535	RW
15	4016		Photocell On Enable	Int	On/Off	RW
16	4017		Photocell Off Enable	Int	On/Off	RW
			Schedule Reference			
17	4018		Panel Number	Int	0 – 99	RW
18	4019		Object Type	Int	0 – 65535 BV= 5 SCH= 17	RW
19	4020		Instance	Word	1 – 65535	RW
20	4021		Schedule Off	Int	On/Off	RW
21	4022		Schedule On	Int	On/Off	RW

22	4023		Astro On Enable	Int	On/Off	RW
23	4024		Astro Off Enable	Int	On/Off	RW
24	4025	BYTE	Astro On Offset	Int	0 – 254	RW
25	4026	LONG	Astro On Calc Hi**	Word	0 – 254	R
26	4027		Astro On Calc Lo**	Word	0 – 254	R
27	4028	BYTE	Astro Off Offset	Int	0 – 254	RW
28	4029	LONG	Astro Off Calc Hi**	Word	0 – 254	R
29	4030		Astro Off Calc Lo**	Word	0 – 254	R
30	4031	BYTE	FlickWarnEnable	Word	1-On / 0-Off	RW
31	4032	BYTE	FlickWarnTime	Word	0 – 240 min	RW
32	4033	WORD	FlickWarnPAEnble	Word	0 – 65535	RW
33	4034	WORD	FlickWarnInputEnb	Word	0 – 65535	RW
		Array of 33 Output Ref/ Switch Ref pairs	**Lighting Outputs**			
			Output Ref			
34	4035		Panel Number	Word	1 – 99	RW
35	4036		Object Type		4 – [BO] 303 – [LG]	RW
36	4037		Object Instance	Word	1 – 65535	RW
			Switch Ref			
37	4038		Panel Number	Word	1 – 99	RW
38	4039		Object Type		3 – [BI] 13 – [MI]	RW
39	4040		Object Instance	Word	0 – 65535	RW
40	4041		Value	Word		RW
41	4042		Switch Value	Word		RW
42	4043 – 4298		32 more entries: 8 registers per entry			RW
298	4299		Common Off Time	Int	0 – 1000	RW
		Array of 16	**Common Groups**			
299	4300		Panel Number	Word	1 – 99	RW
300	4301		Object Type	Word	303 – [LG]	RW
301	4302		Object Instance	Word	1 – 16	RW
302	4303		Value	Word		RW
303	4304 – 4363		15 remote LG refs: 4 registers per entry			RW
			363 registers / LG			

Table 48. Lighting Group setup registers.

Sample Packets

The following are examples of requests to specific setup registers.

Read for LG1 OutOfService Value

Table 49 is an example of read for LG1 OutOfService, Value and Current Priority for Reg4001–Reg4003 (Function 3, LG 1in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	0F	Byte Count	06
Register Address Lo	A0	Register Value Hi (4001)	00
No. of Registers Hi	00	Register Value Lo (4001)	00
No. of Registers Lo	03	Register Value Hi (4002)	00
Packet CRC Hi		Register Value Lo (4002)	02
Packet CRC Lo		Register Value Hi (4003)	00
		Register Value Lo (4003)	06
		Packet CRC Hi	
		Packet CRC Lo	

Table 49. Sample packets for read for Reg4001–Reg4003.

Response is as follows:

Reg4001 (OutOfService) 0x0000 – FALSE,

Reg4002 (Value) 0x0002 – Override

Reg4003 (Current Priority) 0x0006 – Priority 6

Read LG Photocell Reference

Table 50 is an example of read LG Photocell Reference (Object Type, Object Instance and Panel Number) for Reg. 4009–Reg4011 (Function 3, LG1 in map).

The response is Photocell Reference Object Type 1 (AI), Object Instance 2, Panel Number 100.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	0F	Byte Count	06
Register Address Lo	A8	Register Value Hi (4009)	00
No. of Registers Hi	00	Register Value Lo (4009)	64
No. of Registers Lo	03	Register Value Hi (4010)	00
Packet CRC Hi		Register Value Lo (4010)	00
Packet CRC Lo		Register Value Hi (4011)	00
		Register Value Lo (4011)	02
		Packet CRC Hi	
		Packet CRC Lo	

Table 50. Sample packets for read LG photocell reference for Reg4009–4011.

Read Lighting Output Array

Table 51 is an example of read Lighting Output Array Entry 2 Reg4035–Reg4042 (Function 3, LG1 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	0F	Byte Count	10
Register Address Lo	C2	Register Value Hi (4035)	00
No. of Registers Hi	00	Register Value Lo (4035)	64
No. of Registers Lo	08	Register Value Hi (4036)	00
Packet CRC Hi		Register Value Lo (4036)	04
Packet CRC Lo		Register Value Hi (4037)	00
		Register Value Lo (4037)	02
		Register Value Hi (4038)	00
		Register Value Lo (4038)	64
		Register Value Hi (4039)	00
		Register Value Lo (4039)	0D
		Register Value Hi (4040)	00
		Register Value Lo (4040)	66
		Register Value Hi (4041)	00
		Register Value Lo (4041)	00
		Register Value Hi (4042)	00
		Register Value Lo (4042)	02
		Packet CRC Hi	
		Packet CRC Lo	

Table 51. Sample packets for read for lighting output array.

The response to reading Array Index 2 of Lighting Output, which consists of Output Reference and Switch Reference, is as follows:

Reg4035–Reg3037 Output Reference object:
0x006400040002 - 100.BO2

Reg4038–Reg4040 Switch Reference object:
0x0063000E0065 - 100.MI101

Reg4041 Reference Value: 00000

Reg4042 Switch Value: 0002

Single Write OFF to Lighting Group 1

Table 52 is an example of a single write OFF (0x0000) to LG1 Value Reg4002 (Function 6, LG 1in map). Note that OutOfService is automatically set to TRUE with this write.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	06	Function	06
Register Address Hi	0B	Register Address Hi	0B
Register Address Lo	B9	Register Address Lo	B9
Register Value Hi	00	Register Value Hi	00
Register Value Lo	00	Register Value Lo	00
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 52. Sample packets for single write OFF to LG 1.

Read for Lighting Group 1 OutOfService

Table 53 is an example of a read for LG1 OutOfService, Value and Current Priority, Reg4001–Reg4003. This is the result after the single write of the previous example.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	0B	Byte Count	06
Register Address Lo	B8	Register Value Hi (4001)	00
No. of Registers Hi	00	Register Value Lo (4001)	01
No. of Registers Lo	03	Register Value Hi (4002)	00
Packet CRC Hi		Register Value Lo (4002)	01
Packet CRC Lo		Register Value Hi (4003)	00
		Register Value Lo (4003)	01
		Packet CRC Hi	
		Packet CRC Lo	

Table 53. Sample packet for read of LG1 OutOfService.

The response is as follows:

Reg4001 (OutOfService) 0x0001: TRUE

Reg4002 (Value) 0x0001: On

Reg4003 (Current Priority) 0x0001: Priority 1 (Manual Override)

Single Write to Restore OutOfService

Table 54 is an example of a single write to restore the OutOfService to FALSE. We do a write to LG present value with 0x04 (Reg4002).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	06	Function	06
Register Address Hi	0B	Register Address Hi	0B
Register Address Lo	B9	Register Address Lo	B9
Register Value Hi	00	Register Value Hi	00
Register Value Lo	04	Register Value Lo	04
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 54. Sample packets for single write to restore OutOfService.

Read for LG1 OutOfService

Table 55 is an example of a read for LG1 OutOfService, Value and Current Priority, Reg4001–Reg4003. This is the result after the single write of the previous example.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	0B	Byte Count	06
Register Address Lo	B8	Register Value Hi (4001)	00
No. of Registers Hi	00	Register Value Lo (4001)	01
No. of Registers Lo	03	Register Value Hi (4002)	00
Packet CRC Hi		Register Value Lo (4002)	02
Packet CRC Lo		Register Value Hi (4003)	00
		Register Value Lo (4003)	06
		Packet CRC Hi	
		Packet CRC Lo	

Table 55. Sample packets for read for LG1 OutOfService.

The response is as follows:

Reg4001 (OutOfService) 0x0000: FALSE

Reg4002 (Value) 0x0002: Overwrite

Reg4003 (Current Priority) 0x0006: Priority 1 (Default Value)

Schedule

The Controller contains 16 schedule objects that is written by the Modbus Master. The Schedule Event value and OutOfService property can be commanded by Function Codes 05 and 15 and read by Function Code 1.

Schedule Value Registers

The Schedule value registers are listed in Table 56.

Reg.	Contents	Type	Format	Instance
501	SCH01 Value	RW	On/Off	01
502	SCH02 Value	RW	On/Off	02
503	SCH03 Value	RW	On/Off	03
504	SCH04 Value	RW	On/Off	04
505	SCH05 Value	RW	On/Off	05
506	SCH06 Value	RW	On/Off	06
507	SCH07 Value	RW	On/Off	07
508	SCH08 Value	RW	On/Off	08
509	SCH09 Value	RW	On/Off	09
510	SCH10 Value	RW	On/Off	10
511	SCH11 Value	RW	On/Off	11
512	SCH12 Value	RW	On/Off	12
513	SCH13 Value	RW	On/Off	13
514	SCH14 Value	RW	On/Off	14
515	SCH15 Value	RW	On/Off	15
516	SCH16 Value	RW	On/Off	16

Table 56. Schedule value registers.

Sample Packets

Table 57 is an example of a request for SCH Value Reg501–Reg505 (Function 1, SCH1–5 in map). Note that SCH Value Status Reg505–501 (SCH05–SCH01) is shown as hex value 1F or binary 0001 1111, while Reg501 is the LSB and Reg505 is the fifth bit from the left.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	01	Byte Count	01
Register Address Lo	F4	SCH OutOfService 505–501	1F
Quantity of SCH Hi	00	Packet CRC Hi	
Quantity of SCH Lo	05	Packet CRC Lo	
Packet CRC Hi			
Packet CRC Lo			

Table 57. Sample packets for request for SCH Value.

Schedule OutOfService Registers

The Schedule OutOfService registers are listed in Table 58.

Reg.	Contents	Type	Format	Instance
601	SCH01 OutOfService	RW	On/Off	01
602	SCH02 OutOfService	RW	On/Off	02
603	SCH03 OutOfService	RW	On/Off	03
604	SCH04 OutOfService	RW	On/Off	04
605	SCH05 OutOfService	RW	On/Off	05
606	SCH06 OutOfService	RW	On/Off	06
607	SCH07 OutOfService	RW	On/Off	07
608	SCH08 OutOfService	RW	On/Off	08
609	SCH09 OutOfService	RW	On/Off	09
610	SCH10 OutOfService	RW	On/Off	10
611	SCH11 OutOfService	RW	On/Off	11
612	SCH12 OutOfService	RW	On/Off	12
613	SCH13 OutOfService	RW	On/Off	13
614	SCH14 OutOfService	RW	On/Off	14
615	SCH15 OutOfService	RW	On/Off	15
616	SCH16 OutOfService	RW	On/Off	16

Table 58. Schedule OutOfService registers.

Sample Packets

Table 59 is an example of a request for SCH OutOfService status instance Reg601–Reg616 (Function 1, SCH1–16 in map). Note that SCH OutOfService Status Reg4208– Reg4201 is shown as hex value AB or binary 1010 1011, while Reg4201 is the LSB and Reg4208 is the MSB. Reg4201 is SCH01 OutOfService Status.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	01	Function	01
Register Address Hi	02	Byte Count	02
Register Address Lo	58	SCH OutOfService 608–601	AB
Quantity of SCH Hi	00	SCH OutOfService 616–609	AA
Quantity of SCH Lo	10	Packet CRC Hi	
Packet CRC Hi		Packet CRC Lo	
Packet CRC Lo			

Table 59. Sample packets for request for SCH OutOfService status.

Schedule Objects

Each schedule object has a block of registers that are used to maintain the schedule.

The first object starts at register address 20001 and the second object starts at 20291. Table60 indicates the list of schedule objects and their associated registers.Each day has eight pairs of ON/OFF times. Day 1 represents Monday, Day 7 represents Sunday, and Day 8 & 9 represents Calendar day(s). If a day has fewer than eight pairs, all empty pairs are filled with 0xFFFF on its ON/OFF time. Schedule properties can be edited with function code 06 and 16 and read with function code 3. Due to constraints in the code, you can only remove an entry by writing 0xFFFF with function code 06.

Register	SCH Object	Type		Register	SCH Object	Type
20001 – 20290	SCH1	RW		22321 – 22610	SCH9	RW
20291 – 20580	SCH2	RW		22611 – 22900	SCH10	RW
20581 – 20870	SCH3	RW		22901 – 23190	SCH11	RW
20871 – 21160	SCH4	RW		23191 – 23480	SCH12	RW
21161 – 24450	SCH5	RW		23481 – 23770	SCH13	RW
21451 – 21740	SCH6	RW		23771 – 24060	SCH14	RW
21741 – 22030	SCH7	RW		24061 – 24350	SCH15	RW
22031 – 22320	SCH8	RW		24351 – 24640	SCH16	RW

Table 60. Schedule Registers

Note that the first entry of each day's time slot has to be 00:00 to indicate midnight and the time slot's value can be changed but not its time. To schedule 24 hours ON, only one entry is needed; set the midnight value to 0x0001 and it will be turned OFF at midnight the next day. To delete an entry, fill the time and value with 0xFFFF. Only one entry can be deleted at a time. You cannot delete all entries with one write.

Configuration properties are in sets for each Schedule and have the address space shown in Table 61. Each object has 290 registers and there are 16 objects per panel.

Offset	Modbus Reg.	DB size	Property	Format	Range	Type
0	20001	Byte	OutOfService	Int	0–1	RW
1	20002	Byte	Value	Int	0–1	RW
Start Schedule List						
2	20003	Byte	Day 1 Entry 1: Time	Int	00:00–23:59	RW
3	20004	Byte	Value	Int	0 – Off, 1 – On	RW
4	20005	Byte	Day 1 Entry 2: Time	Int	00:00–23:59	RW
5	20006	Byte	Value	Int	0 – Off, 1 – On	RW
6	20007	Byte	Day 1 Entry 3: Time	Int	00:00–24:00	RW
7	20008	Byte	Value	Int	0 – Off, 1 – On	RW
8	20009	Byte	Day 1 Entry 4: Time	Int	00:00–23:59	RW
9	20010	Byte	Value	Int	0 – Off, 1 – On	RW
10	20011	Byte	Day 1 Entry 5: Time	Int	00:00–24:00	RW
11	20012	Byte	Value	Int	0 – Off, 1 – On	RW
12	20013	Byte	Day 1 Entry 6: Time	Int	00:00–23:59	RW
13	20014	Byte	Value	Int	0 – Off, 1 – On	RW
14	20015	Byte	Day 1 Entry 7: Time	Int	00:00–24:00	RW
15	20016	Byte	Value	Int	0 – Off, 1 – On	RW
16	20017	Byte	Day 1 Entry 8: Time	Int	00:00–23:59	RW
17	20018	Byte	Value	Int	0 – Off, 1 – On	RW

Offset	Modbus Reg.	DB size	Property	Format	Range	Type
18	20019	Byte	Day 1 Entry 9: Time	Int	00:00–24:00	RW
19	20020	Byte	Value	Int	0 – Off, 1 – On	RW
20	20021	Byte	Day 1 Entry 10: Time	Int	00:00–23:59	RW
21	20022	Byte	Value	Int	0 – Off, 1 – On	RW
22	20023	Byte	Day 1 Entry 11: Time	Int	00:00–24:00	RW
23	20024	Byte	Value	Int	0 – Off, 1 – On	RW
24	20025	Byte	Day 1 Entry 12: Time	Int	00:00–23:59	RW
25	20026	Byte	Value	Int	0 – Off, 1 – On	RW
26	20027	Byte	Day 1 Entry 13: Time	Int	00:00–24:00	RW
27	20028	Byte	Value	Int	0 – Off, 1 – On	RW
28	20029	Byte	Day 1 Entry 14: Time	Int	00:00–23:59	RW
29	20030	Byte	Value	Int	0 – Off, 1 – On	RW
30	20031	Byte	Day 1 Entry 15: Time	Int	00:00–24:00	RW
31	20032	Byte	Value	Int	0 – Off, 1 – On	RW
32	20033	Byte	Day 1 Entry 16: Time	Int	00:00–23:59	RW
33	20034	Byte	Value	Int	0 – Off, 1 – On	RW
34–289	20035–20290		32 Reg. per day 9 days = 288 Regs (2 for Exceptions – CALs)			

Table 61. Schedule object registers.

Sample Packets

The following are examples of function requests to specific registers.

Read for Schedule OutOfService

Table 62 is an example of a read for SCH OutOfService, Value and Day 1 Entry 1, Reg20001–Reg20004 (Function 3, SCH1in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	4E	Byte Count	06
Register Address Lo	20	Register Value Hi (20001)	00
No. of Registers Hi	00	Register Value Lo (20001)	00
No. of Registers Lo	04	Register Value Hi (20002)	00
Packet CRC Hi		Register Value Lo (20002)	01
Packet CRC Lo		Register Value Hi (20003)	00
		Register Value Lo (20003)	00
		Register Value Hi (20004)	00
		Register Value Lo (20004)	00
		Packet CRC Hi	
		Packet CRC Lo	

Table 62. Sample packets for read of Schedule OutOfService.

The response is as follows:

Reg20001 (OutOfService): 0x0000, FALSE

Reg20002 (Value): 0x0001, ON

Reg20003 (Entry 1, Time): 0x0000, 00:00am

Reg20004 (Entry 1, Value): 0x0000, Off Time

Read for Schedule On and Off Times

Table 63 is an example of a read for SCH Day1 (Monday) Entry 2 OnTime and OffTime Reg20005–Reg20008 (Function 3, SCH1 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	4E	Byte Count	08
Register Address Lo	23	Register Value Hi (20005)	05
No. of Registers Hi	00	Register Value Lo (20005)	00
No. of Registers Lo	04	Register Value Hi (20006)	00
Packet CRC Hi		Register Value Lo (20006)	01
Packet CRC Lo		Register Value Hi (20007)	06
		Register Value Lo (20007)	0F
		Register Value Hi (20008)	00
		Register Value Lo (20008)	00
		Packet CRC Hi	
		Packet CRC Lo	

Table 63. Sample packets for read of Schedule On and Off times.

The response is as follows:

Reg20005–20006: 0x0500, 5:00am; 0x0001, On time

Reg20007–20008: 0x060F, 6:15am; 0x0000, Off time

Read for Schedule Day Entries

Table 64 is an example of a read for SCH Day1 (Monday), all eight entries Reg20003–Reg20034 (Function 3, SCH1 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	4E	Byte Count	40
Register Address Lo	22	Register Value Hi (20003)	00
No. of Registers Hi	00	Register Value Lo (20003)	00
No. of Registers Lo	20	Register Value Hi (20004)	00
Packet CRC Hi		Register Value Lo (20004)	00
Packet CRC Lo		Register Value Hi (20005)	05
		Register Value Lo (20005)	00
		Register Value Hi (20006)	00
		Register Value Lo (20006)	01
		Register Value Hi (20007)	06
		Register Value Lo (20007)	0F
		Register Value Hi (20008)	00
		Register Value Lo (20008)	00
		Register Value Hi (20009)	07
		Register Value Lo (20009)	0F
		Register Value Hi (20010)	00
		Register Value Lo (20010)	01
		Register Value Hi (20011)	09
		Register Value Lo (20011)	00
		Register Value Hi (20012)	00
		Register Value Lo (20012)	00
		Register Value Hi (20013)	0A
		Register Value Lo (20013)	1E
		Register Value Hi (20014)	00
		Register Value Lo (20014	01
		Register Value Hi (20015)	0C
		Register Value Lo (20015)	2D
		Register Value Hi (20016)	00
		Register Value Hi (20016	00
		Register Value Hi (20017)	FF
		Register Value Lo (20017)	FF
		Register Value Hi (20018)	FF
		Register Value Lo (20018)	FF
		Register Value Hi (20019)	FF
		Register Value Lo (20019)	FF
		Register Value Hi (20020)	FF
		Register Value Lo (20020)	FF
		Register Value Hi (20021)	FF
		Register Value Lo (20021)	FF
		Register Value Hi (20022	FF
		Register Value Lo (20022)	FF
		Register Value Hi (20023)	FF
		Register Value Lo (20023)	FF
		Register Value Hi (20024)	FF
		Register Value Lo (20024)	FF
		Register Value Hi (20025)	FF
		Register Value Lo (20025)	FF
		Register Value Hi (20026)	FF
		Register Value Lo (20026)	FF
		Register Value Hi (20027)	FF
		Register Value Lo (20027)	FF
		Register Value Hi (20028)	FF
		Register Value Lo (20028)	FF
		Register Value Hi (20029)	FF
		Register Value Lo (20029)	FF
		Register Value Hi (20030)	FF
		Register Value Lo (20030)	FF
		Register Value Hi (20031)	FF
		Register Value Lo (20031)	FF
		Register Value Hi (20032)	FF
		Register Value Lo (20032)	FF
		Register Value Hi (20033)	FF
		Register Value Lo (20033)	FF
		Register Value Hi (20034)	FF
		Register Value Lo (20034)	FF
		Packet CRC Hi	
		Packet CRC Lo	

Table 64. Sample packets for read of schedule day entries.

The response is as follows:

Reg20003–Reg20004 0x0000: 00:00 AM, 0x0000: Off time

Reg20005–Reg20006 0x0500: 5:00 AM, 0x0001" On time

Reg20007–Reg2008 0x060F: 6:15 AM, 0x0000: Off time

Reg20009–Reg20010 0x070F: 7:15 AM, 0x0001 – On time

Reg20011–Reg20012 0x0900: 9:00 AM, 0x0000: Off time

Reg20013–Reg20014 0x0A1E: 10:30 AM, 0x0001: On time

Reg20015–Reg20015 0x0C2D: 12:45 PM, 0x0000: Off time

…

Reg20017–Reg20018 0xFFFF: No entry

…

Multiple Write Changing a Day Schedule

Table 65 is an example of a multiple write to change the Day1 schedule (Function 16, SCH1in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	10
Register Address Hi	4E	Register Address Hi	4E
Register Address Lo	23	Register Address Lo	23
No. of Registers Hi	00	No. of Registers Hi	00
No. of Registers Lo	0C	No. of Registers Lo	0C
Register Value Hi (20004)	06		
Register Value Lo (20004)	00		
Register Value Hi (20005)	00		
Register Value Lo (20005)	01		
Register Value Hi (20006)	07		
Register Value Lo (20006)	00		
Register Value Hi (20007)	00		
Register Value Lo (20007)	00		
Register Value Hi (20008)	08		
Register Value Lo (20008)	00		
Register Value Hi (20009)	00		
Register Value Lo (20009)	01		
Register Value Hi (20010)	12		
Register Value Lo (20010)	00		
Register Value Hi (20011)	00		
Register Value Lo (20011)	00		
Register Value Hi (20012)	14		
Register Value Lo (20012)	00		
Register Value Hi (20013)	00		
Register Value Lo (20013)	01		
Register Value Hi (20014)	17		
Register Value Lo (20014)	2D		
Register Value Hi (20015)	00		
Register Value Lo (20015)	00		
Packet CRC Hi			
Packet CRC Lo			

Table 65. Sample packets for multiple write changing a day schedule.

Single Write to Remove an Entry

Table 66 is an example of a single write to remove entry 4 (Reg.20008–Reg.20009). Write 0xFFFF to the first entry to remove (Reg.20008).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	06	Function	06
Register Address Hi	4E	Register Address Hi	4E
Register Address Lo	27	Register Address Lo	27
Register Value Hi (20008)	FF	Register Value Hi (20008)	FF
Register Value Lo (20008)	FF	Register Value Lo (20008)	FF
Register Value Hi (20009)	FF	Register Value Hi (20009)	FF
Register Value Lo (20009)	FF	Register Value Lo (20009)	FF
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 66. Sample packets for a single write to remove an entry.

Calendar Objects

Lighting controller contains two calendar objects. A BACNet Calendar object is a list of selected dates. Each date has the format year/month/day/day of week. The first date on the list starts at register address 10004. If the current date in the device object matches an entry in the date list of the Calendar object, then the present value is On; otherwise it is Off. A calendar object can accommodate 90 dates over a 10-year span. Each date entry uses two registers. Table 67 shows the list of calendar and associated registers.

Calendar (CAL) Object – (Registers 10001 – 10366)

Register	SCH Object	Type
10001 – 10183	CAL1	RW
10184 – 10366	CAL2	RW

Table 67. Calendar (CAL) Object Registers

Calendar Object Registers

The calendar object registers are listed in Table 68. Most of the properties can be accessed with Functions 3, 6, and 16. Dates are encoded as follows:

Month = 1–12 (January = 1)

Year = 0–255 (1900–2155)

Register Hi = Year

Register Lo = Month

Day = 1–31

Day of Week = 1–7 (Monday–Sunday)

Register Hi = Day

Register Lo = Day of Week

To add a new entry, read the Date List Count and calculate the next empty slot. For instance, to add Monday May 3, 2004:

The current Date List Count = 3

The next empty slot (Start register of the entry) = (3 x 2) + 10004

Write to Reg10010–Reg10011 = 0x01030568: Monday May 3, 2004

To delete an entry, write FF to the year, month, day, and day of week. For instance: to delete Monday May 3, 2004:

Write to Reg10010–Reg10011 = 0xFFFFFFFF

Note that due to entry reshuffle after a delete, you can only delete one entry by writing to two registers at a time. You can't delete an entry and modify another at the same time.

Sample Packets

The following are examples of function requests to calendar object registers.

Read Calendar Properties OutOfService

Table 69 is an example of a read of CAL properties: OutOfService, Value, Date List Count, first and second entries of the date list: Reg10001–Reg10007 (Function 3, CAL 1 in map).
The response is as follows:

Reg10001 (OutOfService) 0x0000: FALSE,

Reg10002 (Value) 0x0000: OFF

Reg10003 (Date List Count) 0x0003: 3 date entries

Reg10004–Reg10005 first entry: 0x050D0268, Friday, Feb 13, 2004

Reg10006–Reg10007 second entry: 0x010F0368, Monday, Mar 15, 2004

Offset	Modbus Reg.	DB size	Property	Format	Range (Hex)	Type
0	10001	Bit	OutOfService	On/Off	0–1	RW
1	10002	Byte	Value	On/Off	0–1	RW
2	10003	Word	Date List Count		0–90	R
Date List						
3	10004		Day of Week/Day	Int	0x0101–0xFFFF	RW
4	10005		Month/Year	Int	0x0101–0xFFFF	RW
5	10006 –10183 = 89 dates with 2 reg. per date		Total of 183 per CAL			

Table 68. Address space for calendar object registers.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	27	Byte Count	0E
Register Address Lo	10	Register Value Hi (10001)	00
No. of Registers Hi	10	Register Value Lo (10001)	00
No. of Registers Lo	07	Register Value Hi (10002)	00
Packet CRC Hi		Register Value Lo (10002)	00
Packet CRC Lo		Register Value Hi (10003)	00
		Register Value Lo (10003)	03
		Register Value Hi (10004)	05
		Register Value Lo (10004)	0D
		Register Value Hi (10005)	02
		Register Value Lo (10005)	68
		Register Value Hi (10006)	01
		Register Value Lo (10006)	0F
		Register Value Hi (10007)	03
		Register Value Lo (10007)	68
		Packet CRC Hi	
		Packet CRC Lo	

Table 69. Sample packets for a read of CAL properties – OutOfService.

Read Calendar Properties Date List

Table 70 is an example of a read of CAL properties: read the first four entries on the date list: Reg10004–Reg10011 (Function 3, CAL 1in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	27	Byte Count	10
Register Address Lo	10	Register Value Hi (10004)	05
No. of Registers Hi	00	Register Value Lo (10004)	0D
No. of Registers Lo	08	Register Value Hi (10005)	02
Packet CRC Hi		Register Value Lo (10005)	68
Packet CRC Lo		Register Value Hi (10006)	01
		Register Value Lo (10006)	0F
		Register Value Hi (10007)	03
		Register Value Lo (10007)	68
		Register Value Hi (10008)	04
		Register Value Lo (10085)	01
		Register Value Hi (10009)	04
		Register Value Lo (10009)	68
		Register Value Hi (10010)	FF
		Register Value Lo (10010)	FF
		Register Value Hi (10011)	FF
		Register Value Lo (10011)	FF
		Packet CRC Hi	
		Packet CRC Lo	

Table 70. Sample packets for a read of CAL properties – date entries.

The response is as follows:

Reg10004–Reg10005 first entry: 0x050D0268, Friday Feb 13, 2004

Reg10006–Reg10007 second entry: 0x010F0368, Monday Mar 15, 2004

Reg10008–Reg10009 third entry: 0x04010468, Thursday April 1, 2004

Reg10010–Reg10011 fourth entry: 0xFFFFFFFF, empty

Single Write of True to CAL OutOfService

Table 71 is an example of a single write of True to CAL1 OutOfService Reg.10001 (Function 6, CAL 1 in map). The response is that OutOfService is True after the write.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	06	Function	06
Register Address Hi	27	Register Address Hi	27
Register Address Lo	10	Register Address Lo	10
Register Value Hi	00	Register Value Hi	00
Register Value Lo	01	Register Value Lo	01
Packet CRC Hi		Packet CRC Hi	
Packet CRC Lo		Packet CRC Lo	

Table 71. Sample packets for a single write of True to CAL1 OutOfService.

Single Write to CAL1 Date List Count

Table72 is an example of a single write of 5 to the CAL1 Date List Count Reg.10003 (Function 6, CAL1 in map). The response is an error code, since the Date List Count is a read-only property.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	06	Error Code	86
Register Address Hi	27	Exception Code	85
Register Address Lo	12	Packet CRC Hi	
Register Value Hi	00	Packet CRC Lo	
Register Value Lo	05		
Packet CRC Hi			
Packet CRC Lo			

Table 72. Sample packets for a single write to the Date List Count Register.

Multiple Write to Edit Entry in Date List

Table 73 is an example of a multiple write to edit the second entry of the Date List (Function 16, CAL1 in map). This changes the second entry to August 22, 2004: 0x07160868.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	10	Error Code	86
Register Address Hi	27	Register Address Hi	27
Register Address Lo	15	Register Address Lo	15
Register Value Hi	00	Register Value Hi	00
Register Value Lo	02	Register Value Lo	05
Reg. Value 1 Hi	07	Packet CRC Hi	
Reg. Value 1 Lo	16	Packet CRC Lo	
Reg. Value 2 Hi	08		
Reg. Value 2 Lo	68		
Packet CRC Hi			
Packet CRC Lo			

Table 73. Sample packets for a multiple write to edit the Date List.

Read Calendar Properties Date List

Table 74 is an example of a read of CAL properties: read the first four entries on the date list after the new entry was added in the previous example: Reg10004–Reg10011 (Function 3, CAL1 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	27	Byte Count	10
Register Address Lo	13	Register Value Hi (10004)	05
No. of Registers Hi	00	Register Value Lo (10004)	0D
No. of Registers Lo	08	Register Value Hi (10005)	02
Packet CRC Hi		Register Value Lo (10005)	68
Packet CRC Lo		Register Value Hi (10006)	07
		Register Value Lo (10006)	16
		Register Value Hi (10007)	08
		Register Value Lo (10007)	68
		Register Value Hi (10008)	04
		Register Value Lo (10008)	01
		Register Value Hi (10009)	04
		Register Value Lo (10009)	68
		Register Value Hi (10010)	FF
		Register Value Lo (10010)	FF
		Register Value Hi (10011)	FF
		Register Value Lo (10011)	FF
		Packet CRC Hi	
		Packet CRC Lo	

Table 74. Sample packets for a read of CAL properties – date entries.

The response is as follows:

Reg10004–Reg10005 first entry: 0x050D0268, Friday Feb 13, 2004 Reg10006–Reg10007 second entry: 0x07160868, Sunday Aug 22, 2004

Reg10008–Reg10009 third entry: 0x04010468, Thursday April 1, 2004

Reg10010–Reg10011 fourth entry: 0xFFFFFFFF, empty

Multiple Write of New Entry to Date List

Table 75 is an example of a multiple write to add a new entry to the date list (Function 16, CAL1 in map).

Note that since we know that CAL has only three entries, to add a new entry, we add the date to the fourth entry: Reg10010–Reg10011. The new entry added is Wednesday May 5, 2004: 0x03050568.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	10	Function	10
Register Address Hi	27	Register Address Hi	27
Register Address Lo	19	Register Address Lo	19
Register Value Hi	00	Register Value Hi	00
Register Value Lo	02	Register Value Lo	02
Reg. Value 1 Hi	03	Packet CRC Hi	
Reg. Value 1 Lo	05	Packet CRC Lo	
Reg. Value 2 Hi	05		
Reg. Value 2 Lo	68		
Packet CRC Hi			
Packet CRC Lo			

Table 75. Sample packets for a multiple write of a new entry.Read Calendar Properties Date List

Table 76 is an example of a read of CAL properties: read the first four entries on the date list after the new entry in the previous example: Reg10004–Reg10011 (Function 3, CAL1 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	27	Byte Count	10
Register Address Lo	13	Register Value Hi (10004)	05
No. of Registers Hi	00	Register Value Lo (10004)	0D
No. of Registers Lo	08	Register Value Hi (10005)	02
Packet CRC Hi		Register Value Lo (10005)	68
Packet CRC Lo		Register Value Hi (10006)	07
		Register Value Lo (10006)	16
		Register Value Hi (10007)	08
		Register Value Lo (10007)	68
		Register Value Hi (10008)	04
		Register Value Lo (10008)	01
		Register Value Hi (10009)	04
		Register Value Lo (10009)	68
		Register Value Hi (10010)	03
		Register Value Lo (10010)	05
		Register Value Hi (10011)	05
		Register Value Lo (10011)	68
		Packet CRC Hi	
		Packet CRC Lo	

Table 76. Sample packets for a read of CAL properties – date entries.

The response is as follows:

Reg10004–Reg10005 first entry: 0x050D0268, Friday Feb 13, 2004

Reg10006–Reg10007 second entry: 0x07160868, Sunday Aug 22, 2004

Reg10008–Reg10009 third entry: 0x04010468, Thursday April 1, 2004

Reg10010–Reg10011 fourth entry: 0x03050568, Wednesday May 5, 2004

Multiple Write to Delete Entry in Date List

Table 77 is an example of a multiple write to delete the second entry in the date list (Function 16, CAL1 in map).

To delete the second entry, set the date to 0xFFFFFFFF. After the deletion, the third and fourth entries are shifted up. Thus the third becomes the second and the fourth becomes the thidd entry. Because of this you can only delete one entry at a time.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	10	Function	10
Register Address Hi	27	Register Address Hi	27
Register Address Lo	15	Register Address Lo	15
Register Value Hi	00	Register Value Hi	00
Register Value Lo	02	Register Value Lo	02
Reg. Value 1 Hi	FF	Packet CRC Hi	
Reg. Value 1 Lo	FF	Packet CRC Lo	
Reg. Value 2 Hi	FF		
Reg. Value 2 Lo	FF		
Packet CRC Hi			
Packet CRC Lo			

Table 77. Sample packets for a multiple write to delete an entry.

Read Calendar Properties – Date List

Table 78 is an example of a read of CAL properties – read the first four entries on the date list after the entry deletion in the previous example: Reg10004–Reg10011 (Function 3, CAL1 in map).

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	27	Byte Count	10
Register Address Lo	13	Register Value Hi (10004)	05
No. of Registers Hi	00	Register Value Lo (10004)	0D
No. of Registers Lo	08	Register Value Hi (10005)	02
Packet CRC Hi		Register Value Lo (10005)	68
Packet CRC Lo		Register Value Hi (10006)	04
		Register Value Lo (10006)	01
		Register Value Hi (10007)	04
		Register Value Lo (10007)	68
		Register Value Hi (10008)	03
		Register Value Lo (10008)	05
		Register Value Hi (10009)	05
		Register Value Lo (10009)	68
		Register Value Hi (10010)	FF
		Register Value Lo (10010)	FF
		Register Value Hi (10011)	FF
		Register Value Lo (10011)	FF
		Packet CRC Hi	
		Packet CRC Lo	

Table 78. ` Sample packets for a read of CAL properties – date entries.

The response is as follows:

Reg10004–Reg10005 first entry: 0x050D0268, Friday Feb 13, 2004

Reg10006–Reg10007 second entry: 0x04010468, Thursday April 1, 2004

Reg10008–Reg10009 third entry: 0x03050568, Wednesday May 5, 2004

Reg10010–Reg10011 fourth entry: 0xFFFFFFFF, Empty

Compact Event Log

The Lighting Controller uses an object known as the compact event log to store system events. This is a first-in, first-out rotary buffer with a depth adjustable to a maximum of 1500 events. Each event has eight components that can be reported. If the buffer overflows, the oldest event is discarded to make room for the latest. A register is available that contains the number of events in the log. If the log is empty and an attempt is made to read out an event, an exception is returned. Events are read using Function Code 04.

Event Definitions

Table 80 lists the types of events that can be present in the compact event log. Asking for more events than are present in the event log causes an exception. Read register 20001 and 20002 to get the number of events and ask for 12n registers to get the n^{th} number of entries.

Offset	Modbus reg.	DB size	Property	Format	Range	Type
0	20001- 20002	Byte	Total Events	Int	1 – 1500	R
			Alarm Output Ref.			
2	20003	Word	Device No	Word	0 - 65535	R
3	20004	Word	Object Type	Word	0 – 65535	R
4	20005	Word	Object Instance	Word	0 – 65535 BO= 4 BV= 5	R
5	20006	Byte	Alarm Ack	Int	0/1	R
6	20007- 20008	Byte	Alarm Values	Int	0 - 65535, 0 - 65535	R
		Array of 1500	**Event Entries**			
8	20009		Hrs/Mins	Int	0 – 23 | 0 - 59	R
9	20010		Seconds/Hundredths of sec	Int	0 – 59 | 0 - 99	R
10	20011		Week of Day / Day	Int	0 – 07 | 0 – 31	R
11	20012		Month / Year	Int	0 – 12 | 0 – 99	R
12	20013		Event Type	Int	Enum	R
13	20014		Event Argument	Int	Enum	R
			Event Object Ref.			
14	20015		Object Type	WORD	0 – 65535	R
15	20016		Object Instance	WORD	0 – 65535	R
16 – 12007	20017 – 32008		1499 more Event entries (8 registers per entry)			
			12007 registers per CEL			

Table 79. Registers comprising the compact event log.

Each event record is a block of eight registers starting from 20009. Note that these registers are read only. The Alarm Output Ref and Alarm Ack properties are editable. Note that if Alarm Output Reference does not exist, all three registers are filled with 0xFFFF.

Compact Event Log Registers

The registers of the compact event log are listed in Table 79.

Event Type	Number		Event Type	Number
NONE	0		OVER VOLT	80
TIME CHG	58		ALARM	81
DEV ONLINE	59		ALARM ACK	82
DEV OFFLINE	60		BRKR TRIP	83
DEV RESET	61		BRKR TRIP RESTORED	84
DB LOAD	66		PHASE ERR RESTORED	85
DB SAVE	67		OVER VOLT RESTORED	86
DB CLEAR	68		UNDER VOLT RESTORED	87
DEAD BATT	74		DEAD BATT RESTORED	88
PHASE ERR	75		CMD FAIL RESTORED	89
CMD FAIL	76		BRKR SHORTED	90
STATUS ON	77		BRKR SHORT RESTORED	91
STATUS OFF	78		OTHER EVENT	92
UNDER VOLT	79			

Table 80. Event definitions

** Asking for more events than exist will cause an exception. Read register 20001 and 20002 to get number of events and ask for n * 12 registers to get nth number of entries.

Sample Packets

Following are examples of function requests to the compact event log.

Ask for Compact Event Log Event #1

Table 81 is an example of a request for Event#1 Reg20007–Reg20014 (Function 4, CEL1 in map).

The response is as follows:

Reg20007–Reg20008 Time: 16:37:27.60

Reg20009–Reg20010 Date: Wed May 26, 2004

Reg200011 Event Type: 0x003D, Device Reset

Reg20012 Event Argument: 0x0000

Reg20013–Reg20014 Event Object Ref: 0x00090064, DEV100

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	4E	Byte Count	10
Register Address Lo	26	Register Value Hi (20007)	10
Registers Hi	00	Register Value Lo (20007)	25
Registers Lo	08	Register Value Hi (20008)	1B
Packet CRC Hi		Register Value Lo (20008)	3C
Packet CRC Lo		Register Value Hi (20009)	03
		Register Value Lo (20009)	1A
		Register Value Hi (20010)	05
		Register Value Lo (20010)	68
		Register Value Hi (20011)	00
		Register Value Lo (20011)	3D
		Register Value Hi (20012)	00
		Register Value Lo (20012)	00
		Register Value Hi (20013)	00
		Register Value Lo (20013)	09
		Register Value Hi (20014)	00
		Register Value Lo (20014)	64
		Packet CRC Hi	
		Packet CRC Lo	

Table 81. Sample packets for request for compact event log event #1.

Ask for Compact Event Log Total Events

Table 82 is an example of a request for Compact Event Log Total Events, Alarm Output Ref Reg20001–Reg20005.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Function	04
Register Address Hi	4E	Byte Count	06
Register Address Lo	20	Register Value Hi (20001)	00
Registers Hi	00	Register Value Lo (20001)	00
Registers Lo	05	Register Value Hi (20002)	00
Packet CRC Hi		Register Value Lo (20002)	12
Packet CRC Lo		Register Value Hi (20003)	00
		Register Value Lo (20003)	64
		Register Value Hi (20004)	00
		Register Value Lo (20004)	05
		Register Value Hi (20005)	00
		Register Value Lo (20005)	44
		Packet CRC Hi	
		Packet CRC Lo	

Table 82. Sample packets for request for compact event log total events.

The response is as follows:

Reg20001–Reg 20002 Total Event: 0x00000012, 18 decimal

Reg20003–Reg20005 Alarm Output: Ref. 0x006400050044, 100.BO68 (Alarm Relay)

Ask for Compact Event Log Nonexistent Event

Table 83 is an example of a request for Event #21, Reg20173–Reg20180, which does not exist. The response is an error return for invalid data value, since there are only 18 events in the list.

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	04	Error Code	84
Register Address Hi	4E	Exception Code	03
Register Address Lo	CC	Packet CRC Hi	
Registers Hi	00	Packet CRC Lo	
Registers Lo	08		
Packet CRC Hi			
Packet CRC Lo			

Table 83. Sample packets for request for nonexistent event.

Compact Event Log (CEL) – Setup (Registers 60001 – 60008)

Table 84 shows the registers to set up alarm events and alarm output reference.

	Modbus reg.	DB size	Property	Format	Range	Type
0	60001- 60002	Byte	Total Events	Int	1 – 2000	R
			Alarm Output Ref.			
2	60003	Word	Device No	Word	0 - 65535	R
3	60004	Word	Object Type	Word	0 – 65535 BO= 4 BV= 5	R
4	60005	Word	Object Instance	Word	0 – 65535	R
5	60006	Byte	Alarm Ack	Int	0/1	R/W
6	60007- 60008	Byte	**Alarm Values**	Int	0 - 65535, 0 - 65535	R/W

Table 84. registers to set up alarm events and alarm output reference

*Note: Registers 60001 to 60005 must not be written to even though the can be viewed with Function 3.

Alarm Values Enabled	Reg 60007 (Hexadecimal)	Reg 60008 (Hexadecimal)
None	0000	0000
Lighting Alarms,	0000	0800
Lighting Statuses,	1000	0000
Lighting Alarms, Lighting Statuses,	1000	0800
Time Change	0002	0000
Time Change, Lighting Alarms	0002	0800
Time Change, Lighting Statuses,	1002	0000
Time Change, Lighting Alarms, Lighting Statuses	1002	0800
Device Statuses,	2000	0000
Device Statuses, Lighting Alarms,	2000	0800
Device Statuses, Lighting Statuses,	3000	0000
Device Statuses, Lighting Alarms, Lighting Statuses,	3000	0800
Device Statuses, Time Change	2002	0000
Device Statuses, Time Change, Lighting Alarms	2002	0800
Device Statuses, Time Change, Lighting Statuses,	0800	0000
Device Statuses, Time Change, Lighting Alarms, Lighting Statuses	3002	0800

Table 85. possible register values for every Lighting Alarm Value combination

Alarm Acknowledge: Register 60006 will always display 0000 hex. If the Alarm Relay is active writing 0001 hex will acknowledge the Alarm. *When alarm is acknowledged the value of the register will go back to 0000hex.

Alarm Values: Register 60007 – 60008 display the value of the CEL Alarm Values bit list. Table 85 displays all possible register values for every Lighting Alarm Value combination possible in the Compact Event Log.

Load Shedding

The Load Shed object defines the Outputs that will be turned Off, in response to a request for the controller to reduce its the electrical load. Each output can be assigned to one of 5 Shed Levels (Table 87) (to define its priority is relation to the Load Shed request). The lower the priority, the more likely the output is to be turned Off (0 prevents the output from being shed). The Load Shed object properties are read by Function Codes 3/6/16 and written to using Function Codes 6/16. Table 86 lists Load Shed Registers.

Load Shed (LS) object – (Registers 30001 – 30069)

Offset	Modbus Regs	DB size	Property	Format	Range	Type
	30001	BYTE	Value	WORD	0 – 3	R
1	30002	WORD	Shed Duration	WORD	0 – 65535	RW
2	30003	BYTE	Shed Level	WORD	0 -5	RW
			Controlled Object Shed Level			
3	30004	BYTE	BO1 – Shed Level	WORD	0-5	RW
4	30005	BYTE	BO1 – Shed Level	WORD	0-5	RW
	…		…			
68	30069	BYTE	BO66 – Shed Level	WORD	0-5	RW

Table 86. Load shed registers.

Type	Number	Comments
Level 0	0	No Shedding Required
Level 1	1	
Level 2	2	
Level 3	3	
Level 4	4	
Level 5	5	

Table 87. Shed Level definitions

Sample Packets

Request		Response	
Field Name	**Hex**	**Field Name**	**Hex**
Device Address	01	Device Address	01
Function	03	Function	03
Register Address Hi	75	Byte Count	06
Register Address Lo	30	Register Value Hi (30001)	00
Registers Hi	00	Register Value Lo (30001)	00
Registers Lo	03	Register Value Hi (30002)	00
Packet CRC Hi		Register Value Lo (30002)	05
Packet CRC Lo		Register Value Hi (30003)	00
		Register Value Lo (30003)	02
		Packet CRC Hi	
		Packet CRC Lo	

Table88. Simple packets for read of all LS1 properties.

Table 88 is an example of reading all the LS1 properties: Value, Shed Duration, and Shed Level.

The response is as follows:

Reg30001 (Value): 0x0000, Inactive,

Reg30002 (Shed Duration): 0x0005, 5 minutes

Reg30003 (Shed Level): 0x0002, Shed Extraneous

Summary of Registers

Table 89 is a summarized list of all the lighting controller registers.

System Variable	Delta Object	Instance Range	Property(s)	Value range	Func. 1/5/15	Func. 2	Func. 3/6/16	Func. 4
Device registers	Device	1	Various	Various				10001 – 10046
Device Setup	Device	1	Various	Various			50001 – 50046	
Physical Inputs	Multi-state Input (SW)	101 – 116	Value	Enumerated				1 – 16
Physical Inputs	Multi-state Input (SW)	201 – 216	Value	Enumerated				17 – 32
Physical Inputs	Multi-state Input (SW)	301 – 316	Value	Enumerated				33 – 48
Physical Inputs	Multi-state Input (SW)	401 – 416	Value	Enumerated				49 – 64
Physical Inputs	Multi-state Input (SW)	501 – 516	Value	Enumerated				65 – 80
Multi-State Input Setup	Multi-state Input (SW)	101 – 116	Setup	Various			40001 – 40148	
Multi-State Input Setup	Multi-state Input (SW)	201 – 216	Setup	Various			40101 – 40148	
Multi-State Input Setup	Multi-state Input (SW)	301 – 316	Setup	Various			40201 – 40248	
Multi-State Input Setup	Multi-state Input (SW)	401 – 416	Setup	Various			40301 – 40348	
Multi-State Input Setup	Multi-state Input (SW)	501 – 516	Setup	Various			40401 – 40448	
Physical Inputs	Analog Input	1 – 3	Value	Float				101 – 106
Physical Inputs	Analog Input	601 – 608	Value	Float				117 – 132
Physical Inputs	Analog Input	701 – 708	Value	Float				133 – 148
Physical Inputs	Analog Input	1 – 3	Various	Various			1001 – 1027	
Physical Inputs	Analog Input	601 – 608	Various	Various			1101 – 1172	
Physical Inputs	Analog Input	701 – 708	Various	Various			1201 – 1272	
Physical Input Setup	Analog Variable	601 – 608	Value	Float			1301 – 1364	
Physical Input Setup	Analog Variable	701 – 708	Value	Float			1401 – 1464	
System Variable	**Delta Object**	**Instance Range**	**Property(s)**	**Value range**	**Func. 1/5/15**	**Func. 2**	**Func. 3/6/16**	**Func. 4**
Physical Outputs	Binary Output	1 to 66	Value	On/Off	1 – 66			
Outputs Feedback	Binary Output	1 to 66	Feedback Value	On/Off		1-66		
Physical Outputs	Binary Output	1 - 70	OutOfService	On/Off	201 – 270			
Special Outputs	Binary Output	67 – 70	Value	On/Off	67 – 70			
Physical Outputs	Binary Output	1 - 70	Various	Various			2001 – 3260	
Physical Output Override	Binary Variable	1 to 66	Override Value	On/Off			3401 – 3466	
System Variable	**Delta Object**	**Instance Range**	**Property(s)**	**Value range**	**Func. 1/5/15**	**Func. 2**	**Func. 3/6/16**	**Func. 4**

Lighting Group	Lighting Group	1 to 16	Value	On/Off	301 – 316			
Lighting Group	Lighting Group	1 to 16	OutOfService	On/Off	401 - 416			
Lighting Group Setup	Lighting Group	1 to 16	Setup	Various			4001 – 9808	
Schedule	Schedule	1 to 16	Value	On/Off	501 - 516			
Schedule	Schedule	1 to 16	OutOfService	On/Off	601 - 616			
Schedule	Schedule	1 to 16	Setup	Various			20001 – 24640	
Calendar	Calendar	1	Setup	Various			10001 – 10366	
Events	Compact Event Log	1	Event records	Record				20001 to 32008
Events Log Setup	Compact Event Log	1	Event records	Record			60001 – 60008	
Load Shed	Load Shed	1	Various	Various			30001 – 30069	

Table 89.Summary of Registers

GE Consumer & Industrial

General Electric Company
41 Woodford Ave., Plainville, CT 06062

DEH41097 R03 1004