GE Security

Automation Module Protocol

TABLE OF CONTENTS

Revision History

Date Changed	Who Changed	Description
5.23.2000	Rob H, Terry T, Mark H, Dave M	First Revision
7.20.2000	Kris S	Concord updates
9.12.2000	Kris S	Updated for Qubit_b Concord software
10.5.2000	Kris S	Updated for Con25_P1 software
10.26.2000	Kris S	Updated for Con25_P3 software
10.31.2000	Kris S	Added checksum byte
11.16.2000	Kris S	Added backspace token to touchpad display table, communication protocol section, and table of contents
12.06.2000	Kris S	Updated for Con25a software and PAM 0339 software
02.05.2001	Mark H	Advent details added
03.07.2001	Kris S	Updated for PAM 0341 software
04.13.2001	Kris S	Added notes for Concord maximum inbound packet size
04.19.2001	Tim B	Added notes for embedded ACK/NAK in Outbound Messages, Clarified Communication Rules for Behavior
05.07.2001	Tim B	Clarified that message Zone Status (21h) is sent for open (non-restored) zones in response to a Dynamic Data Refresh request.
02.11.2002	Kris S	Added changes for Concord 2.6 and removed Concord Express definitions.
05.07.2002	Kris S	Added changes for Concord 2.53
?	?	?
03.03.2005	Tim B	Removed Concord 2.x, generalized into Advent and Concord, changed to GE document, added reserved commands
05.10.05	Bryan P	Removed reference to the Automation Monitor Program (Automon). We cannot support customer use of this program. Also removed references to PHAST, which is no longer supported
12.15.05	Bryan P	Removed "Confidential" from footer

Definition of Terms (used in the context of this document)

Term	Definition
Automation Device	An aftermarket product intended to monitor/control a Panel.
Automation Module	Interlogix SuperBus 2000 peripheral which provides a gateway between the Panel and the Automation Device.
Autonomous	A device performing an action while not under direct control of another device. The Automation Module does something on its own – not under direct control of the Panel.
Full Duplex	Permits transmission of data in two directions, both ends can transmit at the same time, and the receiver at the opposite end will not have any interference or distortion due to its own transmissions.
Host Processor	The PIC16C77 microcontroller found on the Automation Module. It is in a 44-pin PLCC (Plastic Leaded Chip Carrier).
Inbound	Message originating at the Automation Device and sent to the Panel.
Inbound Buffer	Buffers incoming messages, the size is 82 ASCII characters. See the section titled "Automation Module, Implementation Perspective" for more information.
Last Index	Indicates the number of bytes following the last index byte. The last index is equal to the byte count minus one (in a three-element array, the last index is two).
Outbound	Message originating at the Panel and sent to the Automation Device.
Outbound Buffer	Buffers outgoing messages, the size is 82 binary bytes. See the section titled "Automation Module, Implementation Perspective for more information.
Panel	Interlogix Advent, Concord or future alarm Panel.
Physical Interface	The physical interface between the Automation Module and the Automation Device: is either RS-232 or PHAST. The physical interface between the Automation Module and the Panel is RS-485.
UART Processor	The PIC16C63 microcontroller found on the Automation Module. It is in a 28-pin DIP (Dual In-line Package).

List of Acronyms

Acronym	Definition
ACK	ASCII Acknowledge character (control-F), 06h.
LF	ASCII Linefeed character (control-J), 0Ah.
NAK	ASCII Negative Acknowledge character (control-O), 15h.

Overview

The purpose of this document is to explain the 'how' of the Panel, Automation Module and Automation Device Interaction and provide a list of supported commands.

Interlogix strongly recommends that -- prior to writing any software -- serial port Automation Developers familiarize themselves with the Automation Protocol by reading this document completely, then interacting with the Panel via a terminal emulation application, or serial data monitor to observe outbound message formatting. Note that each outbound message will be retransmitted until it is acknowledged.

Additionally, an almost indispensable tool for software development is a serial data line monitor (either stand-alone, or PC-based.)

Configuration / Physical Interface

Configuring the Automation Module

The Printer/Automation Module is configured at INTERLOGIX for one of the following capabilities: parallel printer, serial communication, or PHAST communication.

The Automation port supports either the proprietary PHAST interface or a standard 3-wire RS-232 interface. From the Panel's point of view, the physical interface is unknown, both are simply "Automation."

Serial Port LED Behavior

- The RS-232 LED will be illuminated during the time an Outbound character from the Automation Module is emitted at the serial port.
- The RS-232 LED is also illuminated for 125 ms each time an Inbound character from the Automation Device is received on the serial port.
- Generally speaking, a quickly flickering LED indicates an Outbound message, whereas a longer flickering indicates an Inbound message.

Serial Port Physical Interface

- Serial parameters: 9600 baud, 8 data bits, odd parity, 1 stop bit.
- Serial Port is a RS-232 level, 3-wire Data Communications Equipment (DCE) serial port interface.
- Use a straight-through cable to connect to a PC's 9-pin serial port.

Female 9-pin D-shell connections as follows:

Pin 2	Transmit data (Automation Module's point of view)
Pin 3	Receive data (Automation Module's point of view)
Pin 5	Ground
Remaining pins	No-connect

Serial Port Protocol

The message consists of four parts sent in the following order:

1. Start of Message
2. Data
3. Checksum
4. Acknowledge or Negative Acknowledge

Start of Message

A single Line Feed (0Ah) is used to signal the start of a message. The line feed character should reset the Host Processor's message parsing pointer as it always indicates the start of a new message.

Data

An 8-bit binary value is sent as two upper-case ASCII digits ('0'...'9', 'A'...'F').

Example:
To expand the 8 bit binary value 3Ch into its ASCII representation:

Send ASCII '3' (33h)
Send ASCII 'C' (43h)

NOTE: Control characters (ACK, NAK, and LF) are sent as a single byte, and not converted into ASCII pairs.

The use of control characters and ASCII data permits software flow control. Only sixteen ASCII characters '0'...'9', 'A'...'F' are used to transfer "data." If binary data were transmitted, the use of hardware flow control, timing specifications, or a byte-stuffing scheme would be required. The control characters and ASCII data protocol was chosen because its advantage in simplicity outweighs the slight loss of efficiency.

Checksum

A checksum is appended to each message. It is the sum of the binary interpretation of all the preceding bytes in the message (control characters and ignored characters excluded), taken modulus 256. The checksum is computed on the 8-bit binary representation of the ASCII pair, rather than the values of the individual ASCII characters. An example checksum calculation, with an overflow, is shown below:

Example
Calculate the checksum for the message: 03h (Last Index), 7Ah, character, and 9Bh. The Last Index is three because the message will contain three bytes: data byte: 7Ah, data byte: 9Bh, and the checksum byte. The checksum calculation is a byte wide, sum of the binary data (excluding the control characters).

Therefore:

Checksum calculation:	***Message***	***Calculation***
Last Index	*03h*	*03h*
First Data Byte	*7Ah*	*7Ah*
Second Data Byte	*9Bh*	*+ 9Bh*
		118h
Checksum (one byte wide)		*18h*

ACK (06h) (Acknowledge)

Sent autonomously by the Automation Device to indicate that the message was well formed (enough bytes were received, and the checksum was correct).

Sent autonomously by the Automation Module to indicate that the message was well formed (enough bytes were received, and the checksum was correct) and that the message has been successfully transmitted to the panel. An ACK sent by the Automation Module does not mean the Panel understood/acted on the message.

An ACK may be issued asynchronously with regard to any Outbound message in progress. This means that an ACK may occur in the middle of an Outbound message.

NAK (15h) (Negative Acknowledge)

Sent autonomously by the Automation Module or Automation Device whenever the message was not well formed (i.e., too many bytes, or invalid checksum). The message originator should re-send its message.

A NAK may be issued asynchronously with regard to any Outbound message in progress. This means that a NAK may occur in the middle of an Outbound message.

Communication Rules for Behavior

1. The Automation Module shall not deliver invalidated messages to the Panel.
   - If the parity of a character is incorrect, that character shall be ignored. Ignoring a character causes a discrepancy in the byte count -- eventually invalidating the message.
   - If the computed checksum does not agree with the received checksum, the message shall be invalidated.
2. Unsolicited messages (i.e., alarms) are sent from the Panel to the Automation Device. If the Automation Device determines an unsolicited message is invalid, it should return a NAK to receive a repeat of the message.
3. When the Automation Module has sent the contents of its input buffer to the Panel, an ACK is generated. An ACK does not mean the Panel understood/acted on the message. The ACK means the message was well formed (enough bytes were received; checksum was correct) and was delivered to the Panel.
4. The Automation Device **shall not** send additional Inbound messages until an ACK or NAK is received from the Automation Module, with the following exception:

- If an ACK or a NAK is not received by the Automation Device within 500ms of sending an Inbound message, something has gone awry. The proper error recovery is to re-send the original Inbound message beginning with the CR/LF.

Note: An ACK or NAK may be issued asynchronously with regard to any Outbound message in progress. This means that an ACK or NAK may occur in the middle of an Outbound message.
Note: The automation module will not process Incoming messages until a prior message has been responded to.

Automation Module, Implementation Perspective

The Automation Module uses two processors (Host and UART) for the serial port protocol. The UART Processor converts parallel data from the Host Processor into serial data, and provides some Inbound message buffering.

Outbound Messages

Outbound message processing is handled by the UART processor in real-time sending characters passed to it by the Host.

The firmware allows the Automation Device to request the last Outbound message be re-sent. The Automation Module handles this request autonomously.

One of the following three things must occur for each Outbound message:

1. ACK. The Automation Device receives a well-formed message and sends the acknowledge character to the Automation Module.
2. NAK. The Automation Device detects a discrepancy in the message and asks for the message to be re-transmitted by sending the negative acknowledge character.
3. Timeout. If the Automation Device sends neither an ACK nor a NAK within 2 seconds, the Automation Module considers this to be an implied NAK and will re-transmit the message.

Inbound Messages

The Automation Module has a 64 character buffer for the **binary** interpretation of Inbound messages from the Automation Device. The LF and checksum are not stored. This means that the maximum byte count for an inbound message is 64 (40h). For example, the largest keypress command that can be sent is: 40 40 01 00 [60 keys] [CS]

NOTE: The maximum byte count for an inbound message to a Concord panel is limited by the panel's internal buffer size. The maximum byte count sent is limited to 58 (3Ah). For example, the largest keypress command that can be sent to a Concord panel is: 3A 40 01 00 [54 keys] [CS].

Action Request Confirmation

The Automation Device will receive feedback from messages sent to the Panel, but the Automation Device may have to request status updates to confirm that a requested action was completed.

Example

The user attempts to arm the security system using an Automation Device

1. *The Automation Device sends a message to the Automation Module to arm the Panel.*
2. *The Automation Module will perform the following tasks:*
   a. *Receives the message*
   b. *Verifies the message is correct*
   c. *Sends the message in Superbus 2000 format to the Panel*
   d. *Sends an ACK the Automation Device*
3. *The Panel interprets the arm the system request based on the following conditions:*
   a. *If the security system has open sensor(s) or other system troubles:*
      i. *The system will not arm*
      ii. *Message/s describing the open sensor(s) or other system troubles will be sent to the Automation Device*
   b. *If the security system does not have open sensor(s) or system troubles*
      i. *The system will arm*
      ii. *Message/s describing arming level will be sent to the Automation Device*

Panel and Automation Device Interaction Example

This example shows the Equipment List Request command and how it would appear to a serial port Automation Device and the Panel.

Example:

Send the Equipment List Request command from the Automation Device to the Panel:

Equipment List command:

02h 02h 04h

Send the following from an Automation Device:

1. *Send the Start of Message information to the Automation Module, ASCII Linefeed character (control-J), 0Ah*

1. *Send the Last Index, 02h*
   a. *Calculate Last Index. The Last Index is the remaining number of bytes (ASCII pairs) to follow in the message. Therefore; the Last Index is 02h (data byte, 02h and the checksum, 04h).*
   b. *High nybble of the Last Index (30h) is sent to the Automation Module.*
   c. *Low nybble of the Last Index (32h) is sent to the Automation Module.*
2. *Send the data byte, 02h*
   a. *High nybble of the data byte (30h) is sent to the Automation Module.*
   b. *Low nybble of the data byte (32h) is sent to the Automation Module.*
3. *Send the checksum, 04h*

   a. Calculate the checksum. The checksum calculation is a byte wide, sum of the hexadecimal representation of each ASCII pair (excluding the control characters).

Checksum calculation:	*High Nybble*	*Low Nybble*	*Hexadecimal Representation*
Last Index	*'0'*	*'2'*	*02h*
Data	*'0'*	*'2'*	*+ 02h*
Checksum (one byte wide)	*'0'*	*'4'*	*04h*

   b. *High nybble of the checksum (30h) is sent to the Automation Module.*
   c. *Low nybble of the checksum (34h) is sent to the Automation Module.*

The 02h 02h 04h message was received by the Automation Module as shown below:

Format	***Start of Message***	***Last Index***		***Data***		***Checksum***	
Hexadecimal	*0Ah*	*30h*	*32h*	*30h*	*32h*	*30h*	*34h*
ASCII	*LF**	*'0'*	*'2'*	*'0'*	*'2'*	*'0'*	*'4'*

***NOTE**: The line feed (LF) character MUST precede the message.*

4. *The Automation Module has received the expected number of message bytes. The Automation Module will decode the ASCII text into the original hexadecimal message and verify the checksum. Then the Automation Module will send the message to the Panel and ACK the Automation Device.*
5. *The Panel receives an Equipment List Request from the Automation Module wrapped in the SuperBus 2000 protocol.*
6. *The Panel will send the Equipment List to the Automation Module, which passes the information to the Automation Device.*

Decimal to Hex to ASCII Conversion List

Decimal	Hexadecimal	ASCII	Key
0	0	NUL	Ctrl @
1	1	SOH	Ctrl A
2	2	STX	Ctrl B
3	3	ETX	Ctrl C
4	4	EOT	Ctrl D
5	5	ENQ	Ctrl E
6	6	ACK	Ctrl F
7	7	BEL	Ctrl G
8	8	BS	Ctrl H
9	9	HT	Ctrl I
10	A	LF	Ctrl J
11	B	VT	Ctrl K
12	C	FF	Ctrl L
13	D	CR	Ctrl M
14	E	SO	Ctrl N
15	F	SI	Ctrl O
16	10	DLE	Ctrl P
17	11	DC1	Ctrl Q
18	12	DC2	Ctrl R
19	13	DC3	Ctrl S
20	14	DC4	Ctrl T
21	15	NAK	Ctrl U
22	16	SYN	Ctrl V
23	17	ETB	Ctrl W
24	18	CAN	Ctrl X
25	19	EM	Ctrl Y
26	1A	SUB	Ctrl Z
27	1B	ESC	Ctrl [
28	1C	FS	Ctrl \
29	1D	GS	Ctrl]
30	1E	RS	Ctrl ^
31	1F	US	Ctrl _
32	20	SP	
33	21	!	
34	22	"	
35	23	#	
36	24	$	
37	25	%	
38	26	&	
39	27	'	
40	28	(	
41	29	)	
42	2A	*	
43	2B	+	
44	2C	,	
45	2D	-	

Decimal	Hexadecimal	ASCII
64	40	@
65	41	A
66	42	B
67	43	C
68	44	D
69	45	E
70	46	F
71	47	G
72	48	H
73	49	I
74	4A	J
75	4B	K
76	4C	L
77	4D	M
78	4E	N
79	4F	O
80	50	P
81	51	Q
82	52	R
83	53	S
84	54	T
85	55	U
86	56	V
87	57	W
88	58	X
89	59	Y
90	5A	Z
91	5B	[
92	5C	\
93	5D	]
94	5E	^
95	5F	_
96	60	`
97	61	a
98	62	b
99	63	c
100	64	d
101	65	e
102	66	f
103	67	g
104	68	h
105	69	i
106	6A	j
107	6B	k
108	6C	l
109	6D	m

46	2E	.	
47	2F	/	
48	30	0	
49	31	1	
50	32	2	
51	33	3	
52	34	4	
53	35	5	
54	36	6	
55	37	7	
56	38	8	
57	39	9	
58	3A	:	
59	3B	;	
60	3C	<	
61	3D	=	
62	3E	>	
63	3F	?	

110	6E	n
111	6F	o
112	70	p
113	71	q
114	72	r
115	73	s
116	74	t
117	75	u
118	76	v
119	77	w
120	78	x
121	79	y
122	7A	z
123	7B	{
124	7C	|
125	7D	}
126	7E	~
127	7F	DEL

Panel to Automation Commands

This section outlines the commands that are sent from the security panel to the Automation Module.

COMMAND: Panel Type (01h)

This command is sent on panel power up initialization and when a communication failure restoral with the Automation Module occurs. Concord also sends the Panel Type in response to a Dynamic Data Refresh Request (20h). Advent also sends the Panel Type in response to a Full Equipment List Request (02h).

Format: 0Bh 01h [PT] [HRh] [HRl] [SRh] [SRl] [SN4] [SN3] [SN2] [SN1] [CS]

Field	Description	Advent	Concord
PT	Panel Type	Commercial Fire 250 0Dh Home Navigator 132 0Fh Commercial Burg 250 10h Home Navigator 250 11h Commercial Burg 500 15h Commercial Fire 500 16h Commercial Fire 132 17h Commercial Burg 132 18h	Concord 14h Express 0Bh Express 4 1Eh Euro 0Eh
HRh	Hardware Revision high	1-26	Will use the 'hh' portion of 57-hhll number[1]
HRl	Hardware Revision low	0-99	Will use the 'll' portion of 57-hhll number[1]
SRh	Software Revision high	0-9	Will use the 'hh' portion of 75-hhll number[2]
SRl	Software Revision low	1-99	Will use the 'll' portion of 75-hhll number[2]
SN4	Serial Number byte 4 (high)	0	Serial number of product (lower right portion bar code)[3]
SN3	Serial Number byte 3	0	Serial number of product (lower right portion bar code)[3]
SN2	Serial Number byte 2	0	Serial number of product (lower right portion bar code)[3]
SN1	Serial Number byte 1 (low)	0	Serial number of product (lower right portion bar code)[3]
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

[1] xxyy=4-digit hardware revision defined as xx=numeric representation of board revision letter a-z; (ex. xx=01=a, 00<xx<27,), yy=the board revision number; (yy=00-09). Example: revision 'C2' would equate to xxyy='0302'

[2] 4-digit firmware set which includes all finished microcontrollers and eeproms

[3] 7-digit board serial number indicating serially the number of the manufactured panel in a product code family

COMMAND: Automation Event Lost (02h) (CONCORD ONLY)

This command is sent if the panel's automation buffer has overflowed resulting in the loss of system events. This command should result in a Dynamic Data Refresh and Full Equipment List request from the Automation Device.

Format: 02h 02h 04h

COMMAND: Clear Automation Dynamic Image (20h)

This command is sent on panel power up initialization and when a communication failure restoral with the Automation Module occurs. The Concord will also send this command when user or installer programming mode is exited.[4]

Format: 02h 20h 22h

[4] This is done instead of sending a message for each item as it is changed (user code deleted, etc.). The Automation Device should perform an Equipment List and Refresh when the Clear Image command is received.

COMMAND: Zone Status (21h)

This command is sent whenever there is a change in zone state (e.g. trip, restore, alarm, cancel, trouble, restoral, bypass, unbypass). Also, if the Automation Module requests a Dynamic Data Refresh Request this command will be sent for each zone that is not normal (i.e. any zone that is open (non restored), in alarm, troubled or bypassed). The remote device should assume that all zones are normal unless told otherwise.

Format: 07h 21h [PN] [AN] [ZNh] [ZNl] [ZS] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8[5]	1-6[6]
AN	Area Number	0-8[7]	0
ZNh	Zone Number high		
ZNl	Zone Number low		1-96
ZS	Zone State	bit 0: 1 = tripped bit 1: 1 = faulted bit 2: 1 = Alarm bit 3: 1 = Trouble bit 4: 1 = Bypassed	(← same)
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

[5] Partition number is 1-8 if partitions are enabled, 0 if areas are enabled.
[6] Could go to 8 in later revisions.
[7] Area number is 1-8 if areas are enabled, 0 if partitions are enabled.

COMMAND: Arming Level (22h/01h)

This command is sent whenever there is a change in the arming level. Also, if the Automation Module requests a Dynamic Data Refresh Request this command will be sent for each partition that is enabled.

Format: 08h 22h 01h [PN] [AN] [UNh] [UNl] [AL] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
UNh	User Number High	0	1 if keyfob 0 if not
UNl	User Number Low	0 – 249 (see table below)	0-252 (see table below) 1-96 for zone number if keyfob
AL	Arming Level	0 = Zone Test 1 = 0ff 2 = Home/Perimeter 3 = Away/Full 4 = Night 5 = Silent	(← same)
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

User Number Description	Advent	Concord
Regular User Codes	6 – 250 or 10 - 250*	00 - 229
Partition *n* Master Code	n	230 - 237
Partition *n* Duress Code	N/A	238 - 245
System Master Code	N/A	246
Installer Code	00	247
Dealer Code	N/A	248
AVM Code	N/A	249
Quick Arm	N/A	250
Key Switch Arm	65535	251
System	65535	252

Note: For user number, UNh byte is 00 and UNl is the user number as listed in the table above. For a keyfob, UNh is 01 and UNl is the zone number for the keyfob.

* First regular user code for 4 partition versions of Advent is 6, for 8 partition versions is 10.

COMMAND: Entry/Exit Delay (22h/03h)

This command is sent whenever an entry or exit delay is started or ended.

Format: 08h 22h 03h [PN] [AN] [DF] [DTh] [DTl] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
DF	Delay Flags	bit 5,4: 00 = standard, 01 = extended, 10 = twice extended bit 6: 1 = exit delay, 0 = entry delay bit 7: 1 = end delay, 0 = start delay	(← same)
DTh	Delay Time high (seconds)		
DTl	Delay Time low (seconds)		
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Alarm/Trouble (22h/02h)

This command is sent to identify alarm and trouble conditions as well as several other system events. Events are specified by three numbers; General Type, Specific Type, and Event Specific Data. The lists below show all the events, categorized by General Type.

Format: 0Dh 22h 02h [PN] [AN] [ST] [SNh] [SNm] [SNl] [GT] [ST] [ESh] [ESl] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
ST	Source Type	0 = bus device 1 = phone 2 = zone 3 = system	0 = bus device 1 = local phone 2 = zone 3 = system 4 = remote phone
SNh	Source Number high	2-byte zone number or 3-byte unit ID	(← same)
SNm	Source Number mid	2-byte zone number or 3-byte unit ID	(← same)
SNl	Source Number low	2-byte zone number or 3-byte unit ID	(← same)
GT	General Type	(see lists below)	(← same)
ST	Specific Type	(see lists below)	(← same)
ESh	Event Specific Data high	(see lists below)	(← same)
ESl	Event Specific Data low	(see lists below)	(← same)
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

Each list below contains the Specific Type codes for the General Type shown in the heading. For example, if GT = 1, and ST = 3, the event is a Police Alarm. Most events do not use the Event Specific Data byte. The underlined events apply only to Advent. For definition of user number, see description in

COMMAND: Arming Level (22h/01h)

Alarm (General Type = 1)

1.0	Unspecified	
1.1	Fire	
1.2	Fire Panic	
1.3	Police	
1.4	Police Panic	
1.5	Medical	
1.6	Medical Panic	
1.7	Auxiliary	
1.8	Auxiliary Panic	
1.9	Tamper	
1.10	No Activity	
1.11	Suspicion	
1.12	Not used	
1.13	Low Temperature	
1.14	High Temperature	
1.15	Keystroke Violation (Touchpad Tamper)	
1.16	Duress	
1.17	Exit Fault	
1.18	Explosive Gas	
1.19	Carbon Monoxide	
1.20	Environmental	
1.21	Latchkey	ES = user number
1.22	Equipment Tamper	
1.23	Holdup	
1.24	Sprinkler	
1.25	Heat	
1.26	Siren Tamper	
1.27	Smoke	
1.28	Repeater Tamper	
1.29	Fire Pump Activated	
1.30	Fire Pump Failure	
1.31	Fire Gate Valve	
1.32	Low CO2 Pressure	
1.33	Low Liquid Pressure	
1.34	Low Liquid Level	
1.35	Entry/Exit	
1.36	Perimeter	
1.37	Interior	
1.38	Near (Two Trip, Concord only)	
1.39	Water Alarm	

Alarm Cancel (General Type = 2)

2.xx xx = same as General Type = 1

Alarm Restoral (General Type = 3)

3.xx xx = same as General Type = 1.

Fire Trouble (General Type = 4)

4.0	Unspecified
4.1	Hardwire
4.2	Ground Fault
4.3	Device
4.4	Supervisory
4.5	Low Battery
4.6	Tamper
4.7	SAM
4.8	Partial Obscurity
4.9	Jam
4.10	Zone AC Fail
4.11	n/u
4.12	NAC Trouble
4.13	Analog Zone Trouble
4.14	Fire Supervisory
4.15	Pump Fail
4.16	Fire Gate Valve Closed
4.17	CO2 Pressure Trouble
4.18	Liquid Pressure Trouble
4.19	Liquid Level Trouble

Fire Trouble Restoral (General Type = 5)

5.yy	yy = same as General Type = 4.

Non-Fire Trouble (General Type = 6)

6.yy	yy = same as General Type = 4.

Non-Fire Trouble Restoral (General Type = 7)

7.yy	yy = same as General Type = 4

Bypass (General Type = 8)

8.0	Direct Bypass	ES = user number
8.1	Indirect Bypass	ES = user number
8.2	Swinger Bypass	
8.3	Inhibit	ES = user number

Unbypass (General Type = 9)

9.0	Direct Bypass	ES = user number
9.1	Indirect Bypass	ES = user number
9.2	Swinger Bypass	
9.3	Inhibit	ES = user number

Opening (General Type = 10)

10.0	Normal Open	ES = user number
10.1	Early Open	ES = user number
10.2	Late Open	ES = user number
10.3	Fail To Open	
10.4	Open Exception	ES = user number
10.5	Open Extension	ES = user number
10.6	Open Using Keyfob/Keyswitch	
10.7	Scheduled Open	
10.8	Remote Open	ES = user number

Closing (General Type = 11)

11.0	Normal Close	ES = user number
11.1	Early Close	ES = user number
11.2	Late Close	ES = user number
11.3	Fail To Close	
11.4	Close Exception	ES = user number
11.5	Close Extension	ES = user number
11.6	Close Using Keyfob/Keyswitch	
11.7	Scheduled Close	
11.8	Remote Close	ES = user number
11.9	Recent Close (Concord only)	ES = user number

Partition Configuration Change (General Type = 12)

12.0	User Access Code Added	ES = user number
12.1	User Access Code Deleted	ES = user number
12.2	User Access Code Changed	ES = user number
12.3	User Access Code Expired	ES = user number
12.4	User Code Authority Changed	
12.5	Authority Levels Changed	
12.6	Schedule Changed	
12.7	Arming or O/C Schedule Changed	
12.8	Zone Added	
12.9	Zone Deleted	

Partition Event (General Type = 13)

13.0	Schedule On	ES = schedule number
13.1	Schedule Off	ES = schedule number
13.2	Latchkey On	
13.3	Latchkey Off	
13.4	Smoke Detectors Reset	
13.5	Valid User Access Code Entered	ES = user number
13.6	Arming Level Changed	ES = user number
13.7	Alarm Reported	
13.8	Agent Release	
13.9	Agent Release Restoral	
13.10	Partition Remote Access	
13.11	Keystroke Violation in Partition	
13.12	Manual Force Arm	ES = user number
13.13	Auto Force Arm	
13.14	Auto Force Arm Failed	
13.15	Arming Protest Begun	ES = arming level attempted
13.16	Arming Protest Ended	ES = current arming level

Partition Test (General Type = 14)

14.0	Manual Phone Test	ES = user number
14.1	Auto Phone Test	
14.2	Auto Phone Test with existing trouble	
14.3	Phone Test OK	
14.4	Phone Test Failed	
14.5	User Sensor Test Started	ES = user number
14.6	User Sensor Test Ended	ES = user number
14.7	User Sensor Test Completed	ES = user number
14.8	User Sensor Test Incomplete	ES = user number
14.9	user Sensor Test Trip	
14.10	Installer Sensor Test Started	
14.11	Installer Sensor Test Ended	
14.12	Installer Sensor Test Completed	
14.13	Installer Sensor Test Incomplete	
14.14	Installer Sensor Test Trip	
14.15	Fire Drill Started	ES = user number

System Trouble (General Type = 15)

15.0	Bus Receiver Failure
15.1	Bus Antenna Tamper
15.2	Main Low Battery
15.3	SnapCard Low Battery
15.4	Module Low Battery
15.5	Main AC Failure
15.6	SnapCard AC Failure
15.7	Module AC Failure
15.8	Aux. Power Failure
15.9	Bus Shutdown

15.10 Bus Low Power Mode
15.11 Phone Line 1 Failure
15.12 Phone Line 2 Failure
15.13 Remote Phone Tamper
15.14 Watchdog Reset
15.15 RAM Failure
15.16 Flash Failure
15.17 Printer Error
15.18 History Buffer (almost) Full
15.19 History Buffer Overflow
15.20 Report Buffer Overflow
15.21 Bus Device Failure
15.22 Failure To Communicate
15.23 Long Range Radio Trouble
15.24 Module Tamper Trouble
15.25 Un-enrolled Module Trouble
15.26 Audio Output Trouble
15.27 Analog Module Trouble
15.28 Cell Module Trouble
15.29 Buddy 1 Failure
15.30 Buddy 2 Failure
15.31 Buddy 3 Failure
15.32 Buddy 4 Failure
15.33 SnapCard Trouble
15.34 Analog Loop Short
15.35 Analog Loop Break
15.36 Analog Address 0
15.37 Un-enrolled Analog Head
15.38 Duplicate Analog Head
15.39 Analog Module Initializing
15.40 Microphone Switch Trouble
15.41 Microphone Trouble
15.42 Microphone Wiring Trouble
15.43 JTECH Premise Paging Trouble
15.44 Voice Siren Tamper Trouble
15.45 Microburst Transmit Failure
15.46 Microburst Transmit Disabled
15.47 Microburst Module Failure
15.48 Microburst Not In Service
15.49 Automation Supervisory Trouble
15.50 Microburst Module Initializing
15.51 Printer Paper Out Trouble

System Trouble Restoral (General Type = 16)

16.xx xx = same as General Type = 15

System Configuration Change (General Type = 17)

17.0	Program Mode Entry	
17.1	Program Mode Exit Without Change	
17.2	Program Mode Exit With Change	
17.3	Downloader Session Start	
17.4	Downloader Session End Without Change	
17.5	Downloader Session End With Change	
17.6	Downloader Error	
17.7	Downloader Connection Denied	
17.8	Date/Time Changed	
17.9	Module Added	
17.10	Module Deleted	
17.11	Speech Tokens Changed	
17.12	Code Changed	
17.13	Panel First Service (cold reset)	
17.14	Panel Back In Service (warm reset)	
17.15	Installer Code Changed	

System Event (General Type = 18)

18.0	Callback Requested	
18.1	Output Activity (not used, see 18.5 & 18.6)	
18.2	Buddy Reception	
18.3	Buddy Transmission Request	
18.4	History Buffer Cleared	
18.5	Output On	ES = output number
18.6	Output Off	ES = output number

COMMAND: Siren Setup (22h/04h)

This command is sent whenever there is something to output with the interior siren output, status beeps, siren sounds, etc. There are 4 cadence bytes. Each bit specifies a 125 mSec time slice, 1=on, 0=off, beginning with the most significant bit of the first cadence byte. The combined cadence is repeated RP times (if RP = 0, the cadence is repeated continuously). The actual outputting of the cadence is not begun until either the Siren Go or the Siren Synchronize command is received (See commands below). If alarm sirens are active, Advent also sends the Siren Setup in response to a Dynamic Data Refresh Request (20h).

Format: 0Ah 22h 04h [PN] [AN] [RP] [CD1] [CD2] [CD3] [CD4] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
RP	Repetition Count	0-255	(← same)
CD1	First Cadence Byte		
CD2	Second Cadence Byte		
CD3	Third Cadence Byte		
CD4	Fourth Cadence Byte		
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Siren Synchronize (22h/05h)

This command is used to synchronize all continuous cadences previously set up by the Siren Setup command.

Format: 03h 22h 05h 2Ah

COMMAND: Siren Go (22h/06h)

This command is used to begin all non-continuous cadences previously set up by the Siren Setup command.

Format: 03h 22h 06h 2Bh

COMMAND: Touchpad Display (22h/09h)

This command sends the touchpad display text tokens to the Automation Module. Advent also sends the Touchpad Display in response to a Dynamic Data Refresh Request (20h).

Format: [LI] 22h 09h [PN] [AN] [MT] [DT] [CS]

Field	Description	Advent	Concord
LI	Last Index	06h + number of display tokens	(← same)
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
MT	Message Type	0 = Normal 1 = Broadcast	(← same)
DT	Display Tokens	See Text Token Table	(← same)
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

Text Token Table

Token	Token Value
0	00
1	01
2	02
3	03
4	04
5	05
6	06
7	07
8	08
9	09
#	0C
:	0D
/	0E
?	0F
.	10
A	11
B	12
C	13
D	14
E	15
F	16
G	17
H	18
I	19
J	1A
K	1B
L	1C
M	1D
N	1E
O	1F
P	20
Q	21
R	22
S	23
T	24
U	25
V	26
W	27
X	28
Y	29
Z	2A
Space	2B
'	2C
-	2D
_	2E
*	2F
AC POWER	30
ACCESS	31
ACCOUNT	32
ALARM	33
ALL	34
ARM	35
ARMING	36
AREA	37
ATTIC	38
AUTO	39
AUXILIARY	3A
AWAY	3B
BACK	3C
BATTERY	3D
BEDROOM	3E
BEEPS	3F
BOTTOM	40
BREEZEWAY	41
BASEMENT	42
BATHROOM	43
BUS	44
BYPASS	45
BYPASSED	46
CABINET	47
CANCELED	48
CARPET	49
CHIME	4A
CLOSET	4B
CLOSING	4C
CODE	4D
CONTROL	4E
CPU	4F
DEGREES	50
DEN	51
DESK	52
DELAY	53
DELETE	54
DINING	55
DIRECT	56
DOOR	57
DOWN	58
DOWNLOAD	59
DOWNSTAIRS	5A
DRAWER	5B
DISPLAY	5C
DURESS	5D
EAST	5E
ENERGY SAVER	5F
ENTER	60
ENTRY	61
ERROR	62

Token	Token Value	Token	Token Value
EXIT	63	NOT	95
FAIL	64	NUMBER	96
FAILURE	65	OFF	97
FAMILY	66	OFFICE	98
FEATURES	67	OK	99
FIRE	68	ON	9A
FIRST	69	OPEN	9B
FLOOR	6A	OPENING	9C
FORCE	6B	PANIC	9D
FORMAT	6C	PARTITION	9E
FREEZE	6D	PATIO	9F
FRONT	6E	PHONE	A0
FURNACE	6F	POLICE	A1
GARAGE	70	POOL	A2
GALLERY	71	PORCH	A3
GOODBYE	72	PRESS	A4
GROUP	73	QUIET	A5
HALL	74	QUICK	A6
HEAT	75	RECEIVER	A7
HELLO	76	REAR	A8
HELP	77	REPORT	A9
HIGH	78	REMOTE	AA
HOURLY	79	RESTORE	AB
HOUSE	7A	RIGHT	AC
IMMEDIATE	7B	ROOM	AD
IN SERVICE	7C	SCHEDULE	AE
INTERIOR	7D	SCRIPT	AF
INTRUSION	7E	SEC	B0
INVALID	7F	SECOND	B1
IS	80	SET	B2
KEY	81	SENSOR	B3
KITCHEN	82	SHOCK	B4
LAUNDRY	83	SIDE	B5
LEARN	84	SIREN	B6
LEFT	85	SLIDING	B7
LIBRARY	86	SMOKE	B8
LEVEL	87	Sn	B9
LIGHT	88	SOUND	BA
LIGHTS	89	SOUTH	BB
LIVING	8A	SPECIAL	BC
LOW	8B	STAIRS	BD
MAIN	8C	START	BE
MASTER	8D	STATUS	BF
MEDICAL	8E	STAY	C0
MEMORY	8F	STOP	C1
MIN	90	SUPERVISORY	C2
MODE	91	SYSTEM	C3
MOTION	92	TAMPER	C4
NIGHT	93	TEMPERATURE	C5
NORTH	94	TEMPORARY	C6

Token	Token Value	Token	Token Value
TEST	C7	WEST	D3
TIME	C8	WINDOW	D4
TIMEOUT	C9	MENU	D5
TOUCHPAD	CA	RETURN	D6
TRIP	CB	POUND	D7
TROUBLE	CC	HOME	D8
UNBYPASS	CD	carriage return	F9
UNIT	CE	pseudo space	FA
UP	CF	carriage return	FB
VERIFY	D0	backspace	FD
VIOLATION	D1	blink next token	FE
WARNING	D2		

COMMAND: Siren Stop (22h/0Bh)

This command is sent to stop any cadence being output.

Format: 05h 22h 0Bh [PN] [AN] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Feature State (22h/0Ch)

This command is sent when a change in feature state occurs and in response to a Dynamic Data Refresh Request. THIS COMMAND IS NOT CURRENTLY SUPPORTED BY ADVENT

Format: [LI] 22h 0Ch [PN] [AN] [FS1] [CS]

Field	Description	Advent	Concord
LI	Last Index	LI = 05h+number of FS bytes	LI = 06h
PN	Partition Number	<not supported>	1-6
AN	Area Number	<not supported>	0
FS1	Feature State Byte 1	<not supported>	Bit 0 = chime Bit 1 = energy saver Bit 2 = no delay Bit 3 = latchkey Bit 4 = silent arming Bit 5 = quick arm **Bit set = on** **Bit clear = off**
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Temperature (22h/0Dh)

This command is sent in response to a Dynamic Data Refresh Request. THIS COMMAND IS NOT CURRENTLY SUPPORTED BY ADVENT

Format: 08h 22h 0Dh [PN] [AN] [TM] [ESL] [ESH] [CS]

Field	Description	Advent	Concord
PN	Partition Number	<not supported>	1-6
AN	Area Number	<not supported>	0
TM	Current Fahrenheit Temperature	<not supported>	20h - 63h
ESL	Energy Saver Low setpoint	<not supported>	2Dh – [ESH-1]
ESH	Energy Saver High setpoint	<not supported>	[ESL+1] – 5Ah
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Time and Date (22h/0Eh)

This command is sent in response to a Dynamic Data Refresh Request. THIS COMMAND IS NOT CURRENTLY SUPPORTED BY ADVENT

Format: 08h 22h 0Eh [HR] [MN] [MM] [DD] [YY] [CS]

Field	Description	Advent	Concord
HR	Hour	<not supported>	00h – 17h
MN	Minute	<not supported>	00h – 3Bh
MM	Month	<not supported>	01h – 0Ch
DD	Day	<not supported>	01h – 1Fh
YY	Year	<not supported>	00h-63h
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Lights State Command (23h/01h)

This command is sent whenever a light state change occurs or in response to a Refresh command. THIS COMMAND IS NOT CURRENTLY SUPPORTED BY ADVENT

Format: [LI] 23h 01h [PN] [AN] [LS1] [LS2] [CS]

Field	Description	Advent	Concord
LI	Last Index	<not supported>	07h
PN	Partition Number	<not supported>	1-6
AN	Area Number	<not supported>	0
LS1	Light State 1	<not supported>	Bit 0 = All Lights Bits 1-7 = Lights 1-7
LS2	Light State 2	<not supported>	Bit 0 = Light 8 Bit 1 = Light 9
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: User Lights Command (23h/02h)

This command is sent whenever a user lights control is made.

Format: 0Bh 23h 02h [PN] [AN] [ST] [SNh] [SNm] [SNl] [LN] [LS] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
ST	Source Type	0 = bus device 1 = phone 2 = zone 3 = system	(← same)
SNh	Source Number high	2-byte zone number or 3-byte unit ID	(← same)
SNm	Source Number mid	2-byte zone number or 3-byte unit ID	(← same)
SNl	Source Number low	2-byte zone number or 3-byte unit ID	(← same)
LN	Light Number	0 = all lights 1-32 specific light	(← same)
LS	Light State	0 = off 1 = on 3 = bright 5 = dim	0 = off 1 = on
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Keyfob Command (23h/03h)

This command is sent whenever the panel receives a keypress from a keyfob.

Format: 08h 23h 03h [PN] [AN] [ZNh] [ZNl] [KC] [CS]

Field	Description	Advent	Concord
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
ZNh	Zone Number high		
ZNl	Zone Number low		1-96
KC	Key Code	0 = disarm 1 = arm 2 = lights 3 = star 4 = arm&disarm 5 = lights&star 6 = long lights 9 = arm&star 10 = disarm&lights	(← same)
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

AUTOMATION TO PANEL COMMANDS

COMMAND: Full Equipment List Request (02h)

Automation Module requests equipment list with this command. Panel sends the following 'Send Equipment List' commands in response to this command. In addition, Advent will send the Panel Type command (01h).

Format: 02h 02h 04h

COMMAND: Single Equipment List Request (02h/[Equipment List Parameter])

Automation Module requests a single equipment list parameter with this command. Panel sends the requested 'Send Equipment List' command(s) in response to this command.

Format: 03h 02h [EP] [CS]

Field	Description	Advent	Concord
EP	Equipment List Parameter	<not supported>	See following table
CS	Checksum	<not supported>	Sum of all preceding bytes in the message, taken modulus 256

EP	Description
03h	Zone Data
04h	Partition Data
05h	SuperBus Device Data
06h	SuperBus Device Capabilities Data
07h	Output Data
09h	User Data
0Ah	Schedule Data
0Bh	Scheduled Event Data
0Ch	Light to Sensor Attachment

COMMAND: Send Equipment List - Zone Data (03h)

This command is sent for each zone programmed, in response to an equipment list request from the Automation Module.

Format: [LI] 03h [PN] [AN] [GN] [ZNh] [ZNl] [ZT] [ZS] [text] [CS]

Field	Description	Advent	Concord
LI	Last Index	LI = 09h if no text programmed	(← same)

PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
GN	Group Number		0-35
ZNh	Zone Number high		
ZNl	Zone Number low		1-96
ZT	Zone Type	0 = sensor 1 = keyfob 2 = touchpad	0 = hardwired 1 = RF 2 = RF touchpad
ZS	Zone Status	bit 0: 1 = tripped[8] bit 1: 1 = fault bit 2: 1 = alarm bit 3: 1 = trouble bit 4: 1 = bypassed	(← same)
text	Zone Text bytes	See Text Token Table	(← same)
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

[8] When sent in response to an equipment list request this bit will always be '0'.

COMMAND: Send Equipment List - Partition Data (04h)

This command is sent for each enabled partition, in response to an equipment list request from the Automation Module.

Format: [LI] 04h [PN] [AN] [AL] [text] [CS]

Field	Description	Advent	Concord
LI	Last Index	LI = 05h if no text programmed	05h
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
AL	Arming Level	0 = Zone Test 1 = 0ff 2 = Home/Perimeter 3 = Away/Full 4 = Night 5 = Silent	1 = Off 2 = Stay 3 = Away 8 = Phone Test 9 = Sensor Test[9]
text	Partition Text bytes	See Text Token Table	<not supported>
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

[9] Either dealer or user sensor test.

COMMAND: Send Equipment List - SuperBus Device Data (05h)

This command is sent for each enrolled bus device, in response to an equipment list request from the Automation Module.

Format: [LI] 05h [PN] [AN] [ID1] [ID2] [ID3] [DS] [UN] [text] [CS]

Field	Description	Advent	Concord
LI	Last Index	LI = 08h if no text programmed	09h
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
ID1	Device ID high		
ID2	Device ID mid		
ID3	Device ID low		
DS	Device Status	0 = ok 1 = failed	0 = ok 1 = failed
UN	Unit Number	<not supported>	0-Fh
text	Device Text	See Text Token Table	<not supported>
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Send Equipment List - SuperBus Device Capabilities Data (06h)

This command is sent for each capability for each enrolled bus device, in response to an equipment list request. For each capability present there will be one byte specifying a capability number and possibly one or more bytes of capability dependent data that consists of the elaboration of the capability.

Format: 07h 06h [ID1] [ID2] [ID3] [CN] [CD] [CS]

Field	Description	Advent	Concord
ID1	Device ID high		
ID2	Device ID mid		
ID3	Device ID low		
CN	Capability Number		
CD	Optional Capability Data		
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

Capability number	Description	Optional data
0x00	Power Supervision	
0x01	Access Control	
0x02	Analog Smoke	
0x03	Audio Listen-In	
0x04	SnapCard Supervision	
0x05	Microburst	
0x06	Dual Phone Line	
0x07	Energy Management	
0x08	Input Zones	Number of inputs
0x09	Phast/Automation/System Manager	
0x0A	Phone Interface	
0x0B	Relay Outputs	Number of outputs
0x0C	RF Receiver	
0x0D	RF Transmitter	
0x0E	Parallel Printer	
0x0F		
0x10	LED Touchpad	
0x11	1-Line/2-Line/BLT Touchpad	
0x12	GUI Touchpad	
0x13	Voice Evacuation	
0x14	Pager	
0x15	Downloadable code/data	
0x16	JTECH Premise Pager	
0x17	Cryptography	
0x18	LED Display	

COMMAND: Send Equipment List - Output Data (07h)

This command is sent for each programmed output in response to an equipment list request.

Format: [LI] 07h [ONh] [ONl] [OS] [ID1] [ID2] [ID3] [ID4] [ID5] [text] [CS]

Field	Description	Advent	Concord
LI	Last Index	LI = 0Ah if no text programmed	LI = 0Ah if no text programmed
ONh	Output Number high		0
ONl	Output Number low		1 - 70
OS	Output State	0=output off 1=output on	Bit 0 = on(1),off(0) Bit 1 = pulse (1)
ID bytes	Defined Below		
text	Optional output text	<not supported>	See Text Token Table
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

Advent

Superbus output ID1-ID3 = device ID
ID4 = output page
ID5 = output bit mask

SnapCard output ID1-ID2 = 0
ID3-ID4 = 1
ID5 = output bit mask

Onboard output ID1-ID2 = 0
ID3 = 1
ID4 = 2
ID5 = output bit mask

LRR output (Advent only) ID1-ID2 = 0
ID3 = 1
ID4 = 3
ID5 = output bit mask

X10 output (Advent only) ID1-ID2 = 0
ID3 = 2
ID4 = X10 module number
ID5 = partition number

COMMAND: Send Equipment List - User Data (09h)

This command is sent for each user in response to an equipment list request.

Format: [LI] 09h [UNh] [UNl] [text] [CS]
OR
Format: [LI] 09h [UNh] [UNl] [UCh] [UCm] [UCl] [text] [CS]

Field	Description	Advent	Concord
LI	Last Index	LI = 04h if no text programmed	04h if no code sent 07h if code sent
UNh	User Number high		0
UNl	User Number low		0 - 252
UCh	User Code high	<not supported>	FFh (unused)*
UCm	User Code mid	<not supported>	00h-99h (BCD)*
UCl	User Code low	<not supported>	00h-99h (BCD)*
text	Optional output text	See Text Token Table	<not supported>
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

* These bytes are only sent if the ACCESS CODE LOCK option is OFF.

COMMAND: Send Equipment List - Schedule Data (0Ah)

This command is sent for each schedule in response to an equipment list request. . THIS COMMAND IS NOT CURRENTLY SUPPORTED BY ADVENT.

Format: [LI] 0Ah [PA] [AA] [SN] [SH] [SM] [PH] [PM] [DY] [SV] [CS]

Field	Description	Advent	Concord
LI	Last Index	<not supported>	0Ah
PA	Partition Assignment	<not supported>	1-6
AA	Area Assignment	<not supported>	0
SN	Schedule Number	<not supported>	0 - 15
SH	Start Hour	<not supported>	0 - 23
SM	Start Minute	<not supported>	0 - 59
PH	Stop Hour	<not supported>	0 - 23
PM	Stop Minute	<not supported>	0 - 59
DY	Days of the Week	<not supported>	Bit 0 = Monday Bit 1 = Tuesday . . bit 6 = Sunday set = on, clear = off
SV	Schedule Validity	<not supported>	<not supported>
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Send Equipment List – Scheduled Event Data (0Bh)

This command is sent for each scheduled event in response to an equipment list request. THIS COMMAND IS NOT CURRENTLY SUPPORTED BY ADVENT.

Note: If an event is not attached to any schedules, it will not be sent in this list.

Format: [LI] 0Bh [PN] [AN] [ET] [S1] [S2] [S3] [S4] [CS]

Field	Description	Advent	Concord
LI	Last Index	<not supported>	07h
PN	Partition Number	<not supported>	1-6
AN	Area Number	<not supported>	0
ET	Event Type	<not supported>	01h-09h = Light 1-9 71h-76h = Output 1-6 E0h = Latchkey Open E1h = Latchkey Close E2h = Exception Open E3h = Exception Close E5h = Auto Arm (level 3)
S1	Schedule Assignment 1	<not supported>	Bit n = schedule n Set = assigned
S2	Schedule Assignment 2	<not supported>	Bit n = schedule n+8 Set = assigned
S3	Schedule Assignment 3	<not supported>	<not supported>
S4	Schedule Assignment 4	<not supported>	<not supported>
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Send Equipment List – Light to Sensor Attachment (0Ch)

This command is sent for each light in response to an equipment list request. THIS COMMAND IS NOT CURRENTLY SUPPORTED BY ADVENT.

Format: [LI] 0Ch [PN] [AN] [L1] … [Ln] [CS]

Field	Description	Advent	Concord
LI	Last Index	<not supported>	0Dh
PN	Partition Number	<not supported>	1-6
AN	Area Number	<not supported>	0
L1-Ln	Sensor number attached to Lights 1 through n	<not supported>	01h-60h 00h=none
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

COMMAND: Equipment List Complete (08h)

This command is sent after all the 03h and 05h commands have been sent in response to an equipment list request.

Format: 02h 08h 0Ah

COMMAND: Dynamic Data Refresh Request (20h)

Automation Module requests data refreshing with this command. Panel sends Arming Level, non-normal Zone Status, Alarm/Trouble Status commands in response. Concord will also respond with the Panel Type (01h), Feature State (22h/0Ch), Temperature (22h/0Dh), Light State (23h/01h), and Time and Date (22h/0Eh) commands. Advent will also respond with the Siren Setup command (22h/04h) if there is an active alarm siren and the Touchpad Display command (22h/09h) to refresh the displays.

Format: 02h 20h 22h

COMMAND: Keypress (40h)

Format: [LI] 40h [PN] [AN] [KP1] … [KPn] [CS]

Field	Description	Advent	Concord
LI	Last Index	4+number of keys	(← same)
PN	Partition Number	0-8	1-6
AN	Area Number	0-8	0
KP1-KPn	Key Press 1 through n	See following table **n < 61**	See following table **n < 55**
CS	Checksum	Sum of all preceding bytes in the message, taken modulus 256	(← same)

00h	0
01h	1
02h	2
03h	3
04h	4
05h	5
06h	6
07h	7
08h	8
09h	9
0Ah	*
0Bh	#
0Ch	Police Panic
0Dh	Aux. Panic
0Eh	Fire Panic
10h	Lights On
11h	Lights Off
12h	Lights Toggle
13h	Keyswitch on
14h	Keyswitch off
15h	Keyswitch toggle (not implemented)
16h	<undefined>
17h	<undefined>
18h	<undefined>
19h	<undefined>
1Ah	<undefined>
1Bh	<undefined>
1Ch	Fire TP – Acknowledge
1Dh	Fire TP – Silence
1Eh	Fire TP – Fire Test
1Fh	Fire TP – Smoke Reset
20h	Keyfob disarm
21h	Keyfob arm
22h	Keyfob lights
23h	Keyfob star
24h	Keyfob arm/disarm
25h	Keyfob lights/star
26h	Keyfob long lights
27h	Keyfob direct arm to level 3
28h	Keyfob direct arm to level 2
29h	Keyfob arm/star
2Ah	Keyfob disarm/lights
2Ch	TP A key
30h	TP B key
2Dh	TP C key
33h	TP D key
2Eh	TP E key
36h	TP F key

Bit 6 = held for a few seconds

COMMAND: Reserved

The following commands are reserved for special use.

Message format:

03 60h xx CS
03h 99h xx [CS]
03h 98h xx [CS]