A Sierra Monitor Company

Driver Manual

(Supplement to the FieldServer Instruction Manual)

FS-8700-45 Grinnell TFX Minerva

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after October 2009

Driver Version:	1.03
Document Revision:	1

TABLE OF CONTENTS

1 Grinnell TFX Minerva Description 3

2 Driver Scope of Supply 3

2.1 Supplied by FieldServer Technologies for this driver 3

2.2 Provided by the Supplier of 3rd Party Equipment 3

2.2.1 Hardware 3

2.2.2 Required 3rd Party Software 3

2.2.3 Required 3rd Party Configuration 3

3 Hardware Connections 4

3.1 Hardware Connection Tips / Hints 4

4 Data Array Parameters 5

5 Configuring the FieldServer as a Grinnell TFX Minerva Passive Client 6

5.1 FieldServer Title and Panel Number 6

5.2 Client Side Connection Descriptions 7

5.3 Client Side Node Descriptors 7

5.4 Client Side Map Descriptors 8

5.4.1 FieldServer Related Map Descriptor Parameters 8

5.4.2 Driver Related Map Descriptor Parameters 8

5.4.3 Timing Parameters 8

5.4.4 Map Descriptor Example 1 - Monitoring the TLT-530 interface's network fault status 9

5.4.5 Map Descriptor Example 2 - Monitoring points on a TFX panel and storing all new point event states 10

5.4.6 Example 3 - Monitoring points on a TFX panel and storing only specific point event states 10

5.4.7 Map Descriptor Example 4 - Generating Point Events on the TFX network 11

Appendix A. Useful Features 12

Appendix A.1. Write-through mode 12

Appendix A.2. Node timeout value 12

Appendix A.3. Connection Inter-character Timeout 12

Appendix B. Reference 13

Appendix B.1. Bus Channel 13

Appendix B.2. Loop 13

Appendix B.3. AlarmState 13

Appendix B.4. Network Fault Status Byte Interpretations 16

1 GRINNELL TFX MINERVA DESCRIPTION

The Grinnell TFX Minerva driver allows the FieldServer to transfer data to and from devices over RS-232 using Grinnell TFX Minerva protocol. The Grinnell driver can emulate either a server or client in order to initiate the sending of messages or receiving messages, although the FieldServer itself should be considered a server.

Max Nodes Supported

FieldServer Mode	Nodes	Comments
Server	62	The FieldServer can monitor point events from a total of 62 TFX panels.

2 DRIVER SCOPE OF SUPPLY

2.1 Supplied by FieldServer Technologies for this driver

FieldServer Technologies PART #	Description
FS-8917-02	RJ45 to DB9F connector adapter
FS-8917-01	RJ45 to DB25M connection adapter
FS-8917-21	RS-485 connection adapter

2.2 Provided by the Supplier of 3rd Party Equipment

2.2.1 Hardware

Part #	Description
-	Grinnell TFX System
-	Grinnell TLT-530 Third Party Interface Module
976291	Grinnell Cable

2.2.2 Required 3rd Party Software

Any software to configure the TLT-530 or attached TFX panels.

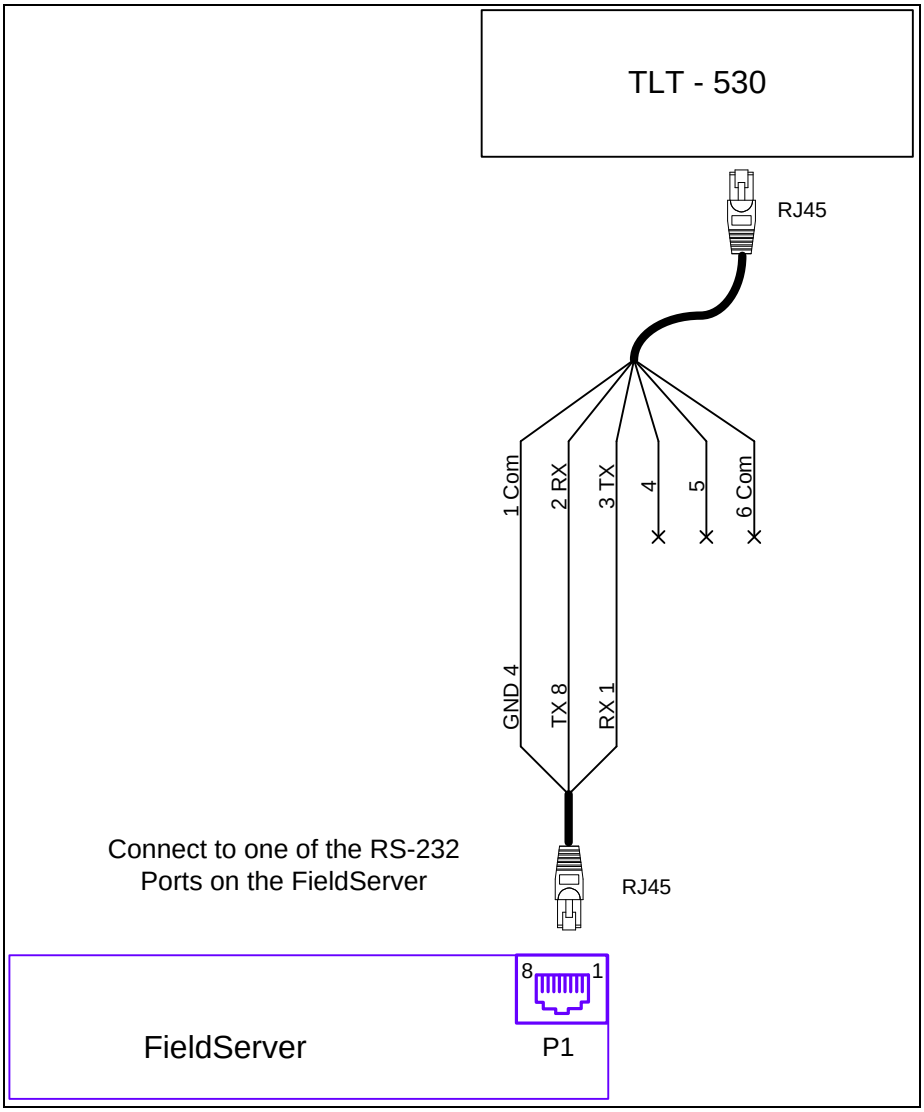
2.2.3 Required 3rd Party Configuration

The TLT-530 should be set up for the correct level mode that will allow monitoring of network panel point events and if desired, allow the FieldServer to generate network panel point events.

3 HARDWARE CONNECTIONS

The FieldServer is connected to the TLT-530 Third Party Interface Module as shown in the connection drawing below.

Configure the TLT-530 Third Party Interface Module according to manufacturer's instructions.



3.1 Hardware Connection Tips / Hints

The FieldServer plugs into a TLT-530 board installed in a TFX 500/800 fire alarm panel.

The FieldServer appears as a Node on the Grinnell Network.

4 DATA ARRAY PARAMETERS

Data Arrays are "protocol neutral" data buffers for storage of data to be passed between protocols. It is necessary to declare the data format of each of the Data Arrays to facilitate correct storage of the relevant data.

Section Title		
Data_Arrays		
Column Title	Function	Legal Values
Data_Array_Name	Provide name for Data Array	Up to 15 alphanumeric characters
Data_Array_Format	Provide data format. Each Data Array can only take on one format.	Float, Bit, UInt16, SInt16, Packed_Bit, Byte, Packed_Byte, Swapped_Byte
Data_Array_Length	Number of Data Objects. Must be larger than the data storage area required by the Map Descriptors for the data being placed in this array.	1-10, 000

Example

```
//  Data Arrays
Data_Arrays
Data_Array_Name   , Data_Format   , Data_Array_Length
Network_Fault     , Byte          , 1
Point_St_Values   , Byte          , 2
Point_States      , Bit           , 190
```

5 CONFIGURING THE FIELDSERVER AS A GRINNELL TFX MINERVA PASSIVE CLIENT

For a detailed discussion on FieldServer configuration, please refer to the FieldServer Configuration Manual. The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer (See ".csv" sample files provided with the FieldServer).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate on a Grinnell TFX Minerva fire panel network with the use of a TLT-530 interface module.

The Grinnell TFX Minerva protocol is a symmetrical Point to Point link layer protocol. There is no set master/slave relationship between the Grinnell FieldServer driver and the devices on the fire panel network, although a fully configured FieldServer could be considered a Grinnell TFX Minerva server since it allows clients external to the FieldServer to receive point event state information and generate point events on the fire panel network. Since the FieldServer Grinnell driver passively listens for events as opposed to actively polling for events such as would be the case in a master/slave protocol, the driver is said to operate as a passive client when monitoring points. In passive client mode, the driver can receive but not generate events on the fire panel network. In server mode, the driver can only generate events on the fire panel network. It will be shown in subsequent sections that the driver can be used in a combination of both modes if necessary.

5.1 FieldServer Title and Panel Number

TFX panels on the fire panel network each have a unique panel number from 1 to 62. The Grinnell driver needs a panel number as well to be able to receive and generate events on the network. The FieldServer's panel number is specified with the System Station Address parameter:

Section Title		
FieldServer		
Column Title	**Function**	**Legal Values**
Title	Provide name for FieldServer	Up to 39 alphanumeric characters
System_Station_Address	FieldServer Panel Number on network	1 – 62

Example

```
FieldServer
Title,                , System_Station_Address
Grinnell Client,      , 3
```

5.2 Client Side Connection Descriptions

Section Title		
Connections		
Column Title	**Function**	**Legal Values**
Port	Specify through which port the device is connected to the FieldServer	P1-P8, R1-R2[1]
Protocol	Specify protocol used	Grinnell
Baud*	Specify baud rate	4800 or **9600** (default on TLT-530)
Parity*	Specify parity	**None**
Data_Bits*	Specify data bits	**8**
Stop_Bits*	Specify stop bits	**1**
Poll _Delay*	Time between internal polls	0-32000 seconds, **1 second**
IC_Timeout	T2 protocol intercharacter timeout value.	0.02s

Example

```
//      Client Side Connections

Connections
Port    , Protocol    , Baud    , Parity    , Handshaking    , IC_Timeout
P1      , Grinnell    , 9600    , None      , None           , 0.02s
```

5.3 Client Side Node Descriptors

Section Title		
Nodes		
Column Title	**Function**	**Legal Values**
Node_Name	Provide name for node	Up to 32 alphanumeric characters
Node_ID	0 –Used to obtain network fault status of a locally connected TLT-530 255 –Broadcast Node_ID used to address all panels with one message. 1-62 –Used to address a single TFX panel on the network.	0; 255; 1-62
Protocol	Specify protocol used	Grinnell
Port	Specify through which port the device is connected to the FieldServer	P1-P8, R1-R2[1]
Timeout	Timeout value to use on Node	≥ 0s, a value of 10s is recommended

[1] Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

Example

```
//  Client Side Nodes

Nodes
Node_Name    , Node_ID   , Protocol   , Port   , Timeout
Local        , 0         , Grinnell   , P1     , 10s
Broadcast    , 255       , Grinnell   , P1     , 10s
Panel04      , 4         , Grinnell   , P1     , 10s
```

5.4 Client Side Map Descriptors

5.4.1 FieldServer Related Map Descriptor Parameters

Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from Section 4.
Data_Array_Offset	Starting location in Data Array	0 to (Data_Array_Length -1) as specified in Section 4.
Function	Function of Client Map Descriptor	Rdbc, Passive_Client, Wrbx, Wrbc, Passive

5.4.2 Driver Related Map Descriptor Parameters

Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from	One of the node names specified in "Client Node Descriptor" above
Bus_Channel	Specifies the channel number within the panel	The Bus_Channel number has to be entered in hex.Refer to Appendix B.1 for more information.
Loop	Specifies the loop number within the channel	The Loop number has to be entered in hex. Refer to Appendix B.2 for more information.
Point	The point number on the loop and associated channel	Addressable loops: 1 – 127 Local, remote buses: 0 – 80
AlarmState*	Specifies the new event point state for a point this Map Descriptor will react to	Refer to Appendix B.3 for more information.
Zone	Zone value used when generating point events	0 – 255
Group	Group value used when generating point events	1 – 511
Sector	Sector value used when generating point events	0 – 80
Length	The number of consecutive points from "Point" this Map Descriptor points to.	

5.4.3 Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Rate at which the network fault location is polled	≥0.001s

5.4.4 Map Descriptor Example 1 - Monitoring the TLT-530 interface's network fault status

The Client driver allows an actively polling Map Descriptor to monitor the TFX network health. This Map Descriptor periodically reads the TFX network's fault status. The TLT-530 monitors the attached TFX network and returns a fault status message if polled. A Scan_Interval of 10s is recommended to to provide a good indication of network health without cluttering the network. This active Map Descriptor is typically used when it is necessary to maintain link integrity with the TLT-530. The FieldServer will report the link going offline when the TLT-530 stops responding to the polling Map Descriptor.

```
Node_Name   , Node_ID   , Protocol   , Port
Local       , 0         , Grinnell   , P1
```

The node specified here must be declared with a node ID of zero for the driver to recognize it as the attached TLT-530

```
Map_Descriptor_Name   , Scan_Interval   , Data_Array_Name   , Data_Array_Offset   , Function   , Node_Name   , Length
CMD001                , 10s             , Network_Fault     , 0                  , Rdbc       , Local       , 1
```

A scan interval of 10s will poll the TLT-530 once every 10s to maintain link integrity and monitor network health.

The TFX network's fault status will be stored in this data array. See the network fault status byte table for details.

This is an active Map Descriptor polling the TLT-530 once every scan interval seconds.

Alternatively or in addition the following Server Map Descriptor could be used which will be updated with a network fault value from the TLT-530 should the network's fault status change. Note that the use of only a Server Map Descriptor to monitor network fault status will result in a loss of link integrity monitoring since the update message is sent asynchronously from the TLT-530 when the network's fault status change.

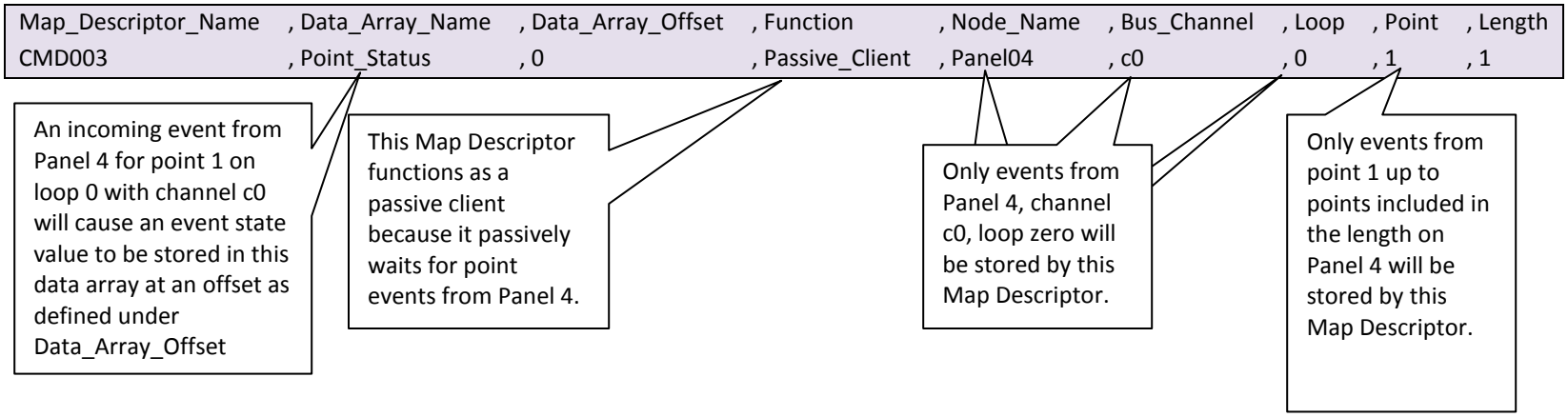
```
Map_Descriptor_Name   , Data_Array_Name   , Data_Array_Offset   , Function   , Node_Name   , Length
CMD002                , Wr_Netw_Fault     , 0                  , Passive    , Local       , 1
```

Refer to Appendix B.4 for information on interpreting Network Error Status bytes.

5.4.5 Map Descriptor Example 2 - Monitoring points on a TFX panel and storing all new point event states

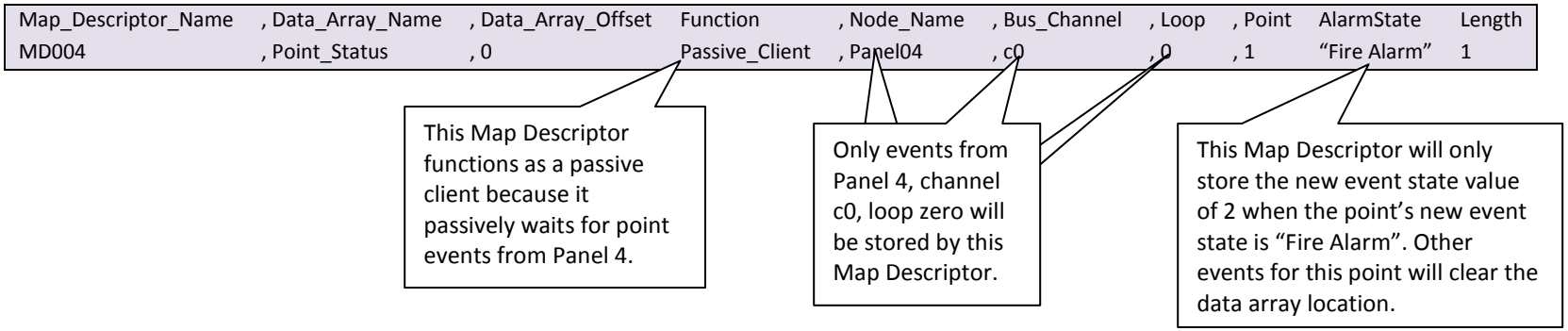
A Map Descriptor must be defined for a single point or a range of points on a specific TFX panel, bus channel and loop. Note that incoming events from panels that have no Map Descriptors defined will be ignored so it is important to ensure that Map Descriptors are defined for all points that have to be monitored.

The Map Descriptor defined below will store the event state value for points starting from Point up to points included in the Length. In this case only one point is monitored. Refer to Appendix B.3 for the event state values that will be stored.



5.4.6 Example 3 - Monitoring points on a TFX panel and storing only specific point event states

The Map Descriptor defined below with an AlarmState of "Fire Alarm" will only store the new event state of 2 when the incoming event for a point it matches has a new event state of "Fire Alarm". Conversely, if an event arrives for the same point with a new event state value other than (2 or "Fire Alarm"), the data array location will be cleared to indicate that the point is no longer in the "Fire Alarm" state.



5.4.7 Map Descriptor Example 4 - Generating Point Events on the TFX network

To generate point events on the network, Map Descriptors with a write-type function have to be used. A Wrbx function Map Descriptor may be used to generate an event when the Data Array location referenced by the Map Descriptor is written to. A Wrbc function Map Descriptor may be used to continuously generate an event every scan interval seconds. When using a Wrbc function Map Descriptor, a scan interval value of 10s or greater is recommended to prevent flooding the communications channel with messages.

5.4.7.1 Extended Event Messages

Extended Event Messages are sent in broadcast mode to all panels on the network. A node ID of 255 on a Map Descriptor will cause a message to be sent in broadcast mode. The Broadcast's Node_ID must be defined as 255 under Section 5.3

Map_Descriptor_Name	, Data_Array_Name	, Data_Array_Offset	, Function	, Node_Name	, Bus_Channel	, Loop	, Point	, Group	, Sector	, Zone	, Length
CMD005	, Write_Status	, 0	, Wrbx	, Broadcast	, c0	, 0	, 1	, 1	, 2	, 3	, 1

The WRBX function will only cause an event message when the data array location is written to.

The broadcast node ID must be defined to be 255 under Nodes.

5.4.7.2 Event Log Messages

Event log messages are sent to a specific panel on the network. A node ID of 1 to 62 on a Map Descriptor will cause a message to be sent directly to the panel with number the same as the node ID.

When writing an event value into the data array at data array offset, the following Map Descriptor will cause an event log message to be sent to Panel with number 4. (provided Panel04 's node ID was defined to be 4 as well).

Map_Descriptor_Name	, Data_Array_Name	, Data_Array_Offset	, Function	, Node_Name	, Bus_Channel	, Loop	, Point	, Group	, Sector	, Zone	, Length
CMD006	, Write_Status	, 0	, Wrbx	, Panel04	, c0	, 0	, 1	, 1	, 2	, 3	, 1

The Wrbx function will only cause an event message when the data array location is written to.

The event log message will be sent to the panel with Node ID defined for Panel04.

Appendix A. Useful Features

Appendix A.1. Write-through mode

The Map Descriptor examples on monitoring points on a TFX network uses a Map Descriptor function called "Passive_Client". These Map Descriptors point to Data Arrays that essentially serve the data to a Client external to the FieldServer. Should a write (or read) Map Descriptor of another protocol driver be used to write the events to a server external to the FieldServer, the Map Descriptor function must be changed to Passive which will allow the write-through of the updated value. Normally, write-throughs are not allowed on "Passive_Client" Map Descriptors since this would cause a write-through on the Grinnell driver to occur which would cause an event message to be generated which will be sent back to the network.

Appendix A.2. Node timeout value

Wher event messages are passed from the FieldServer onto the network or from the network to the FieldServer, these messages pass through the TLT-530. During increased network activity or noisy communication conditions transmission retries may occur between network panels and the TLT-530 delaying the delivery of the message to the FieldServer. A timeout value of at least 10s is recommended to prevent inadvertent timeouts on the FieldServer before the message has been received.

Appendix A.3. Connection Inter-character Timeout

The TLT-530 may be silent for up to 20ms during the transmission of a message to the FieldServer. The Grinnell TFX Minerva protocol recommends an inter-character timeout value of 20ms to be used to prevent the FieldServer timing out on the reception of a valid message.

Appendix B. Reference

Appendix B.1. Bus Channel

The Bus_Channel field in the Map Descriptor specifies the channel within the panel. The bus channel in the Map Descriptor must match the channel reported in the events from the panel before a Map Descriptor will react to incoming events. The Bus Channel value is entered in hex. The top three bits of the channel indicate the type of bus:

Channel #	Description
000	Local Bus
001	Intercontroller Bus
010	Remote Bus
011	Network Addresses 0-31
100	Network Addresses 32-62
110	Virtual Channel

For virtual channels, the rest of the bits are 0. For other channels, the bottom 4 bits specify which channel on the bus. For example, the addressable loops for a TFX-800 are driven by ALXM sub-processors and are identified as being Local Buses 11 through 15 (0B to 0F). The two addressable loops on a TFX-500 are driven from the CPU and are identified as a virtual channel C0.

Appendix B.2. Loop

The loop field in the Map Descriptor specifies the loop within the channel. Loops are also specified in hex. Use the following table to determine the loop:

Loop # (hex)	Description
0	First Addressable Loop
40	Second Addressable Loop
80	Expansion Bus point type
81	Real point type
82	Pseudo point type
83	Keyboard point type
84	Addressable point type
85	User Number type

Appendix B.3. AlarmState

The optional AlarmState field may be used to specify a specific new event state for points. This field causes the Map Descriptor to only store the specified event state when an event is received for which the point's new state is the same as the AlarmState. All events received with a new point state different than AlarmState will cause the Map Descriptor to clear the associated data array location. The table below lists the legal string values for AlarmState with the associated event state values.

AlarmState	EventState	AlarmState	EventState
Fire Alert	1	Point Isolated	96
Fire Alarm	2	Signalling Isolated	97
Fire Evac	3	Zone Deisolated	98
Call Point Alert	4	Point Deisolated	99
Call Point Evac	5	Signalling Deisolated	100
Full Evacuate	6	Data Error 1	101
Input Active	7	Data Error 2	102
Loop Failure	8	Data Error 3	103
Probation Finished	9	Data Error 4	104
Intruder Alarm	10	Data Error 5	105
Intruder raid	11	Intruder Shunt Activated	106
Intruder 24 Hour	12	Intruder Shunt Off	107
Set Timeout	13	Ram Fault	108
Ground Fault	14	Checksum Failure	109
Waterflow Alarm	15	Clear	110
Fire Drill	16	Clear From Fault	111
Supervisory Fault	17	Point Tested 2	112
Enable Sig.Delay	18	Clear Attention Reqd.	113
Disable Sig.Delay	19	Vesda Airflow Fault	114
Medical Alert	20	Ram Overflow	115
Isolate Warning	21	Fault Overflow	116
Coincidence Warning	22	Earth Fault	117
Delay Started	23	Config. Finished	118
Shunt Special	24	Config. Failed	119
Fire Prealarm	25	Printer Offline	120
Alarm Warning	26	Printer Online	121
System Isolate	27	Fire Door Closed	122
System De-Isolate	28	Fire Door Open	123
Part Set	29	Fire Door Failed	124
Tamper	30	Set	125
Tamper O/C	31	Unset	126
Tamper S/C	32	Reset	127
Tamper Lo Id	33	System Powered Up	128
Tamper Hi Id	34	Operator Log On	129
Tamper No Response	35	Operator Log Off	130
Password; Too Many Tries	36	Manager Log On	131
Alarm Verifying	37	Manager Log Off	132
Zonal Unset	38	Engineer Log On	133
Zonal Set	39	Engineer Log Off	134
Premises Unoccupied	40	Clock Cleared	135
Premises Occupied	41	Clock Set	136
Weekly Test Reminder	42	System Silence	137
Fire Test Alarm	43	Input On	138
Auto Test Alarm	44	Input Off	139
Probation Alarm	45	Point Unset	140
Walk Test Started	46	General Fault Log	141
Test Finished	47	Entry Alarm	142
Point Tested	48	Intruder Fault	143
Noisy Sensor	49	Group Active	144
Fault - No Response	50	Mnau Fault Event	145

AlarmState	EventState	AlarmState	EventState
Fault - Unconf'd Device	51	Non-Logging Clear	146
Fault - Hi Id	52	Non-Logging Fault	147
Fault - Lo Id	53	Saturn Group Isolated	148
Fault - Device	54	Saturn Group Connected	149
Fault - Invalid Cond.	55	Input Active Alarm	150
Fault - Lo L.T.A. Cond.	56	Input Active (not logged)	151
Fault - Hi L.T.A. Cond.	57	Group Enable	152
Fault - O/C	58	Group Disable	153
Fault - S/C	59	Group Deactivate	154
Fault - A/D Comp.1	60	Group Reset	155
Fault - A/D Comp.2	61	Auxiliary Alarm	156
Fault - A/D Comp.3	62	Not Set Warning	157
Addr Loop Noise Warning	63	Not Set Alarm	158
Output Stuck	64	Group Purge Delays	159
Fault - Board	65	24v Too Hi Fault	160
Fault - Cp Wiring S/C	66	Activation Counter	161
Fault - Cp Wiring O/C	67	Event Resound	162
Fault - Dm Wiring	68	Reset Button	163
Fault - Psu Wiring O/C	69	No Panel Response	164
Fault - Psu Wiring S/C	70	Response From UNP	165
Fault - Wiring Fault 1	71	Net Module Fault	166
Fault - Wiring Fault 2	72	Net Common Fault	167
Fault - Lbus	73	Net Port Left Fault	168
Fault - Rbus	74	Net Ground Fault	169
Net Comms Fault	75	Net Data Transfer Fault	170
Fault - Rnet	76	Net Ring Fault	171
Fault - Alxm Psu	77	Net No Panel Supervision	172
Fault - Battery	78	Net Port Right Fault	173
Fault - Charger	79	Alarm Clear	174
Fault - Aux Charger	80	Supervisory Cleared	175
Fault - Mains	81	Non Logging Trouble	176
Battery Critical	82	Detector Con Low Warning	177
Fault - Sounder	83	Detector Con High Warning	178
Fault - Sounder Line O/C	84	User Event #1	239
Fault - Sounder Line S/C	85	User Event #2	240
Fault - Sounder Eol	86	User Event #3	241
O/P Control Failure	87	User Event #4	242
Xbus Failure	88	User Event #5	243
Display Fault	89	User Event #6	244
General Fault	90	User Event #7	245
Relay Coil Fault	91	User Event #8	246
Wiring Short Circuit	92	User Event #9	247
Wiring Open Circuit Fault	93	User Event #10	248
Configuration Fault	94	User Event #11	249
Zone Isolated	95	User Event #12	250

Appendix B.4. Network Fault Status Byte Interpretations

Bit 0 – 7	Description	Bit 0 – 7	Description
1000 0000	Data Transfer Fault caused by missed broadcast	0000 1000	Network Ground Fault
0100 0000	Net Comms Fault	0000 0100	Port Right Fault
0010 0000	Ring Continuity Fault	0000 0010	Port Left Fault
0001 0000	Data Transfer Fault	0000 0001	Common Fault