# Multicrest FACP Serial Driver FS-8705-25

Chipkin - Enabling Integration

## TABLE OF CONTENTS

# 1   Multicrest FACP Driver Description

The Multicrest FACP protocol driver can be used to connect to suitably enabled Multicrest FACP panels from Nohmi Bosai Ltd.

The Multicrest driver is a passive client driver intended for connection to the serial port of a Multicrest Fire Alarm Control Panel (FACP). A passive client driver waits for messages to be sent to it (by the panel). The driver cannot send messages to the panel and hence it cannot request the state of any point in the panel.

The driver can process alarm and trouble events, system troubles and some other messages from the panel.

The driver can only be used as a client. Minimal server functionality is provided only to support our ongoing quality assurance program by facilitating automated testing of the driver. Server operation is not documented or supported.

**Synchronization**

A consequence of the fact that this is a passive client driver, is that the FieldServer must be synchronized with the panel by restarting the FieldServer and then resetting the panel to reset the data.

**Max Nodes Supported**

| FIELDSERVER MODE | NODES | COMMENTS |
|---|---|---|
| Passive Client | 1 | Only one panel can be connected to a single FieldServer serial port. Multiple networked panels can be supported through this connection to one panel. |
| Active Server (Simulate a Panel) | 0 | Not supported or documented. |

## 2 Driver Scope of Supply

### 2.1 Supplied with this driver

| FIELDSERVER TECHNOLOGIES PART # | DESCRIPTION |
|---|---|
| 8915-10 | No specific cables are shipped with this driver.<br><br>A generic RJ45 Ethernet cable is shipped with the hardware. |
| FS-8705-21 | Driver Manual. |

## 3 Hardware Connections

Multiple WorkStation protocols and connection supported. See list of FieldServer Drivers.

### 3.1    Hardware Connections

Connect the RS232 cable to TB4  as shown.
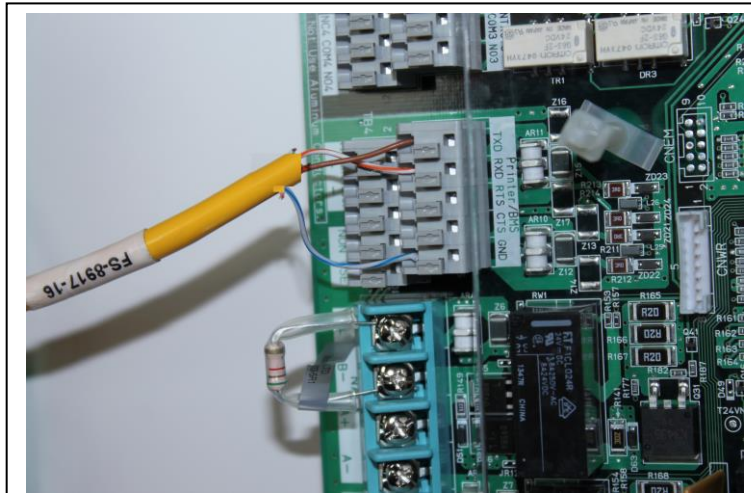


Printer/BMS port on TB4 of the PCA-N3060-PSU card

Blue/White

Brown

Orange/White
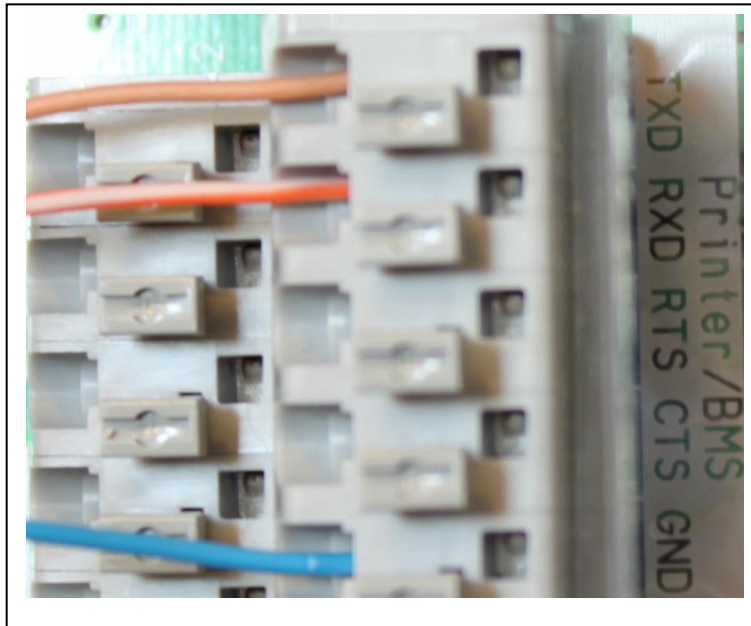
Connect RJ45 end to P2 or P2 on **FS30 Series.** Connect pigtail end to TB4 on fire panel.
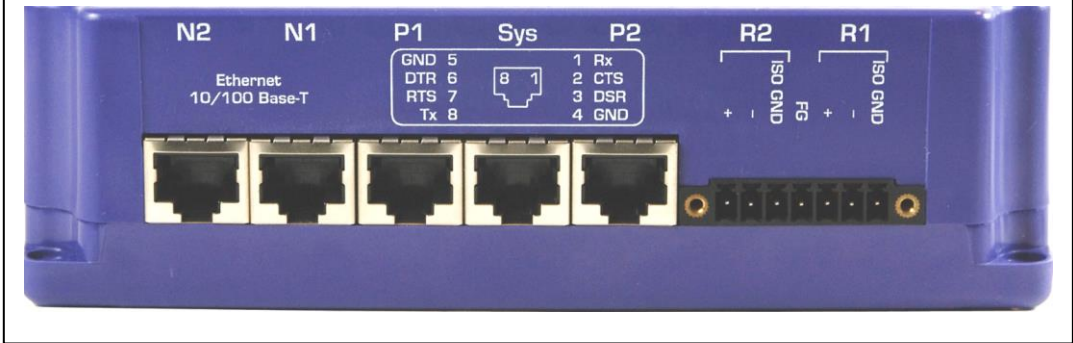
**QuickServer Connections**

The other end of this cable connect to TB4 on the fire panel. That connection is shown in other images.

## 3.2    Block Diagram
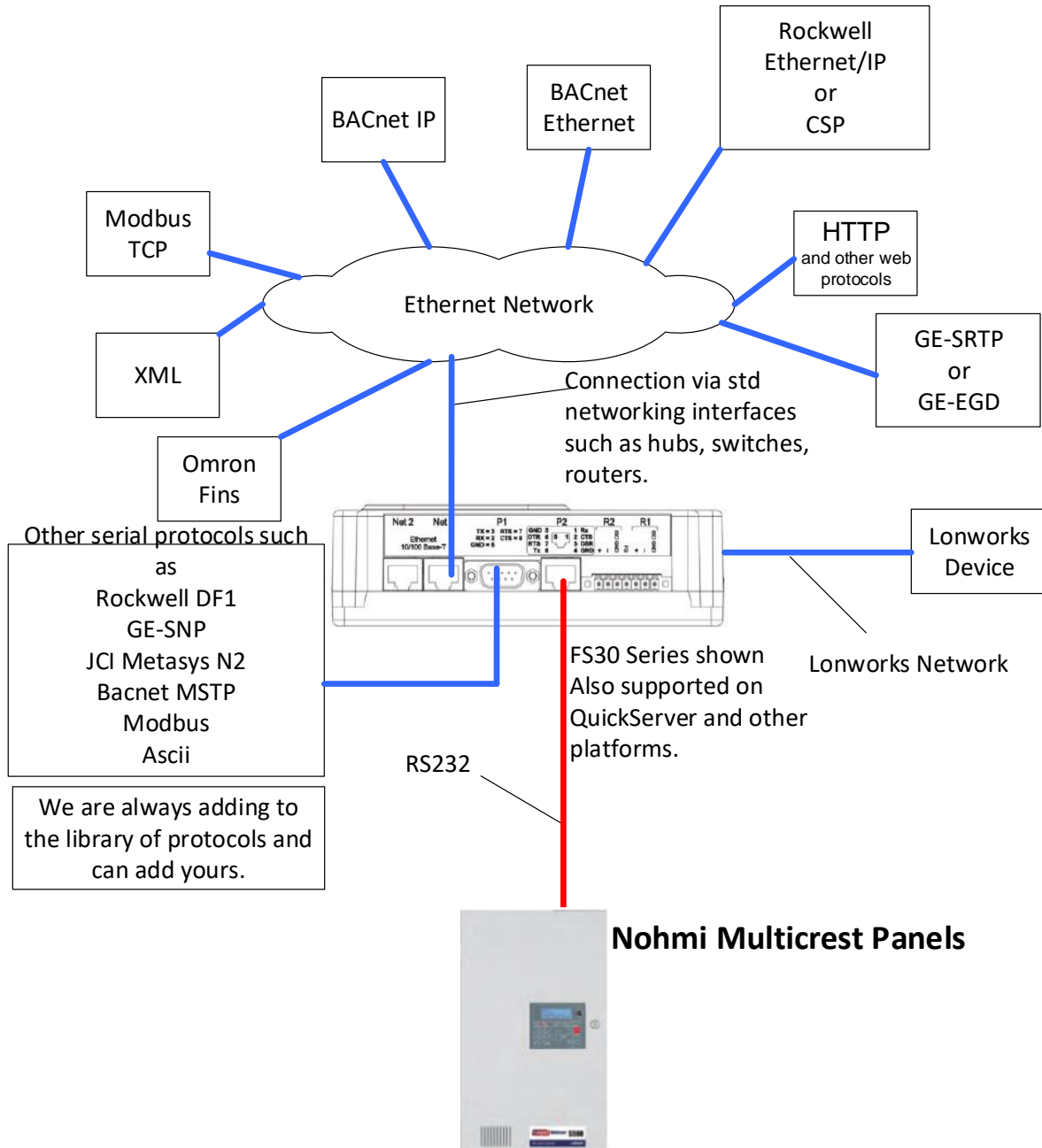
### 3.3    Recommended Cable Assembly

An RJ45 to Terminal cable can be constructed by cutting the end off a Ethernet patch cable (not a cross over cable) as shown below. This is used to connect the FieldServer RS-232 port (RJ-45) to the Multicrest FACP. Note that some FieldServer models have terminal blocks for this connection instead of RJ-45, and for this a standard terminal to terminal wiring setup would apply.
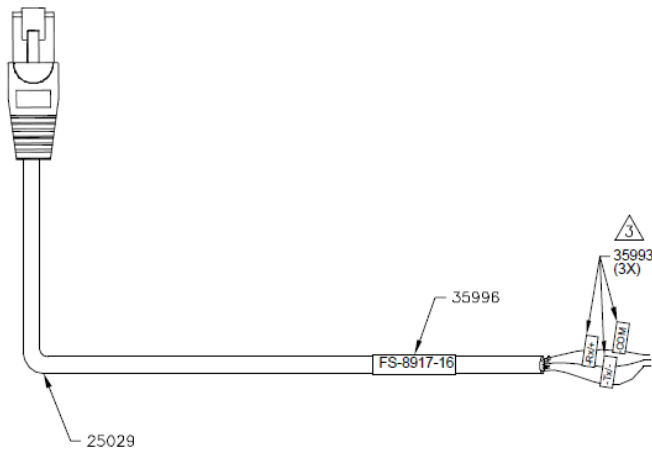


NOTES:    UNLESS OTHERWISE SPECIFIED:

1.    CUT OFF ONE END OF CABLE.

2.    CUT JACKET APPROX. 1.5".
   KEEP 3 WIRES: BRONW, BLUE/WHITE, ORANGE/WHIE. STRIP
   AND TIN THE ENDS.

3.    LABEL WIRES:

   LABEL "FS−RX/+" FOR BROWN WIRE − RS45−01

   LABEL "FS−COM" FOR BLUE/WHITE WIRE − RS45−04

   LABEL "FS−TX/−" FOR ORANGE/WHITE WIRE − RS45−08

# 4    Configuring the FieldServer as a Multicrest 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 FS).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with an Multicrest  system.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for Multicrest monitoring, the driver independent FieldServer buffers need to be declared in the "Data Arrays" section, the destination device addresses need to be declared in the "Client Side Nodes" section, and the data required from the servers needs to be mapped in the "Client Side Map Descriptors" section. Details on how to do this can be found below.

Note that in the tables, * indicates an optional parameter, with the bold legal value being the default.

### 4.1   Data Arrays

| 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. | **Recommended**: Bit, UInt16, <br><br> Also Supported: Float, Uint32, 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 |

### 4.1.1   Data Arrays – Example

```
//   Data Arrays
Data_Arrays
Data_Array_Name,          Data_Format,          Data_Array_Length,
MulticrestStats,          UNT16,                200
```

## 4.2    Client Side Connections

Create one connection for each Multicrest serial port. Each connection can only be used to connect to a single Multicrest interface/port.

| SECTION TITLE | | |
|---|---|---|
| Connections | | |

| COLUMN TITLE | FUNCTION | LEGAL VALUES |
|---|---|---|
| Port | Specify which port the device is connected to the FieldServer | P1-P8 |
| Protocol | Specify protocol used | Multicrest |
| Baud* | Specify baud rate | Driver Supports : 110; 300; 600; 1200; 2400; 4800; **9600**; 19200; 28800; 38400; 57600 Baud<br><br>*Multicrest* supports:  9600 |
| Data_Bits * | Specify parity | Driver Supports : 7,**8**<br><br>*Multicrest* supports:  7 |
| Stop_Bits* | Specify data bits | Driver Supports : **1**,2<br><br>*Multicrest* supports:  1 |
| Parity * | Specify stop bits | Driver Supports : Odd, Even, **None**<br><br>*Multicrest* supports:  Even |

### 4.2.1    Client Side Connection Descriptions – Example

```
//      Client Side Connections



Connections


Port,                 Baud,                 Parity,        Protocol, Data_Bits , Stop_Bits


P1,                   9600,                 Even,          Multicrest , 7           , 1
```

## 4.3   Client Side Nodes

Create one Node per FACP in the network only.

| SECTION TITLE |
| --- |
| Nodes |

| COLUMN TITLE | FUNCTION | LEGAL VALUES |
| --- | --- | --- |
| Node_Name | Provide name for node | Up to 32 alphanumeric characters<br><br>**NB !** The name does not need to correspond to the Node name configured in the Multicrest system. |
| Node_ID | Station address of physical server node<br><br>This parameter is not used directly by the driver. We recommend that a unique Node ID's be given to each node. | 0-64<br><br>Corresponds to the Node numbers of panels. Master node is typically zero. |
| Protocol | Specify protocol used | Multicrest |

### 4.3.1   Client Side Nodes – Example

```
//   Client Side Nodes


Nodes

Node_Name,            Node_ID,            Protocol,            Connection

MainPanel,            0,                  Multicrest ,        P1
```

**4.4    Client Side Map Descriptors**

**4.4.1    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 "Data Array" section above |
| Data_Array_Offset | Starting location in Data Array | 0 to maximum specified in "Data Array" section above |
| Function | Function of Client Map Descriptor.. | Passive |

**4.4.2    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 |
| Data_Type | This commonly used parameter is not used by this driver. | |
| Length | Length of Map Descriptor

Reserves space in the Data Array. | 1,2,3….

Set to the value of the maximum device number on the loop.

For system troubles etc. set the length to 1 |
| Address | This commonly used FieldServer parameter is not used by this protocol. | |

| | | |
|---|---|---|
| Multicrest_Function | Tell the driver what kind of event will be stored using this Map Descriptor.<br><br>Eg. If you want 'Alert' events stored set the Multicrest_Function =Alert | System_Trouble<br>Alert<br>Action<br>Confirm<br>Alarm<br>AlarmF1<br>AlarmF2<br>Verify<br>Supervisory<br>Active<br>DischargeCnf<br>PreDischarge<br>DischargeX<br>Device_Fault<br>Device_Mismatch<br>Device_Bypass<br>Missing_Device<br>Multiple_Add<br>Low_Temp<br>Sensor_Fault<br>Device_Dirty<br>Fault<br>Panel_Trouble<br>Classification<br>The following 2 items need to be used in a special way.<br>Any_Active<br>Any_Trbl |
| Multicrest_Addr | Loop and Circuit Number in one field. | 1:1<br>1:2 |

| | | |
|---|---|---|
| | The format is loop:Circuit<br><br>Eg loop 1 circuit 2 is specified as 1:2 | 1:3<br><br>1:4<br><br>2:1<br><br>etc |
| Multicrest_Loop | If you are using the parameter<br><br>'Multicrest_Addr" then this paramters should not be specified.<br><br>If Multicrest_Function = System_Trouble or Classification then this parameter should be specified with a dash or omitted.<br><br>For other functions set the value of the parameter to the loop number whose events you want stored. | -, 1 , 2, …. |
| Multicrest_Circuit | If you are using the parameter<br><br>'Multicrest_Addr" then this paramters should not be specified.<br><br>If Multicrest_Function = System_Trouble or Classification then this parameter should be specified with a dash or omitted.<br><br><br>When events are processed, in looking for a storage location,, the driver looks to match the circuit number.<br><br>For other functions set the value of the parameter to | 1,2,3,4 |

| | the loop number whose events you want stored. | |
|---|---|---|
| Multicrest_Summary_DA | A single device on a single loop can in be in multiple states simultaneously. For example, a Smoke detector can report an alert, an action or an alarm event. Normally these events are stored in different locations / Data Arrays. If you want a summary for a device then use this parameter.<br><br>Read more in section **Error! Reference source not found. Error! Reference source not found.** | The Name of a Data Array defined in the Data Array section. Or a dash if not required. |
| Multicrest_MultiAlarm_DA | A single device on a single loop can in be in multiple states simultaneously.<br><br>Use this parameter to specify the name of a Data Array to be used to store a value which reports which is the most important activated Alarm state. | The Name of a Data Array defined in the Data Array section. Or a dash if not required. |
| Multicrest_MultiTrbls_DA | Use this parameter to specify the name of a Data Array to be used to store a value which reports which Trouble is active. | The Name of a Data Array defined in the Data Array section. Or a dash if not required. |

## 4.5   How Data is Stored / Summarizing Events

### 4.5.1   Classification

When a device on a loop reports an event, the driver learns the device classification (think of the classification as the device type. Eg. Analog Smoke Detector). The classification information is not cleared when a system reset occurs. Notes on the Classifications and a table of Classifications and their corresponding enumeration is provided in an appendix to this manual.

## 4.5.2   System/Panel Events

System/Panel events are those reported by the main/networked panels as opposed to a single addressable device. One Map Descriptor is required to store System events for main panel and one more for each networked panel in the system. System/Panel Trouble data is cleared when there is a System Reset.

The offset into the data array is used to indicate the event type. Eg.  A "Remote Annunciator #01 Trouble" event will be indicated when offset 87 it set to 1 The table below provides a full list of system troubles and the offset number. The table can be updated by the configuration engineer.

On the next pages a table is provided which can be used to identify the Data Array offset used by each type of System or Panel Trouble event.

Table of System/Panel Troubles and offset

| System Trouble Event | Data Array Offset | System Trouble Event | Data Array Offset |
|---|---|---|---|
| Configuration Data Error | 1 | SLC 07 Open | 26 |
| PSU 1 AC Power Fault | 2 | SLC 08 Open | 27 |
| PSU 2 AC Power Fault | 3 | SLC 09 Open | 28 |
| PSU 1 Battery Fault | 4 | SLC 10 Open | 29 |
| PSU 2 Battery Fault | 5 | SLC 11 Open | 30 |
| PSU 1 Ground Fault | 6 | SLC 12 Open | 31 |
| PSU 2 Ground Fault | 7 | SLC 01 Short | 32 |
| SCU 1 Ground Fault | 8 | SLC 02 Short | 33 |
| SCU 2 Ground Fault | 9 | SLC 03 Short | 34 |
| SCU 3 Ground Fault | 10 | SLC 04 Short | 35 |
| SCU 4 Ground Fault | 11 | SLC 05 Short | 36 |
| SCU 5 Ground Fault | 12 | SLC 06 Short | 37 |
| SCU 6 Ground Fault | 13 | SLC 07 Short | 38 |
| PSU 1 24V Fault | 14 | SLC 08 Short | 39 |
| PSU 2 24V Fault | 15 | SLC 09 Short | 40 |
| PSU 1 Charging Ciruit Fault | 16 | SLC 10 Short | 41 |
| PSU 2 Charging Ciruit Fault | 17 | SLC 11 Short | 42 |
| PSU 1 Board Trouble | 18 | SLC 12 Short | 43 |
| PSU 2 Board Trouble | 19 | SLC 01 Power Fault | 44 |
| SLC 01 Open | 20 | SLC 02 Power Fault | 45 |
| SLC 02 Open | 21 | SLC 03 Power Fault | 46 |
| SLC 03 Open | 22 | SLC 04 Power Fault | 47 |
| SLC 04 Open | 23 | SLC 05 Power Fault | 48 |
| SLC 05 Open | 24 | SLC 06 Power Fault | 49 |
| SLC 06 Open | 25 | SLC 07 Power Fault | 50 |

Table of System/Panel Troubles and offset

| System Trouble Event | Data Array Offset | System Trouble Event | Data Array Offset |
|---|---|---|---|
| SLC 08 Power Fault | 51 | OCU 8 Trouble | 76 |
| SLC 09 Power Fault | 52 | OCU 1 Type Error | 77 |
| SLC 10 Power Fault | 53 | OCU 2 Type Error | 78 |
| SLC 11 Power Fault | 54 | OCU 3 Type Error | 79 |
| SLC 12 Power Fault | 55 | OCU 4 Type Error | 80 |
| SLC 01 Trouble | 56 | OCU 5 Type Error | 81 |
| SLC 02 Trouble | 57 | OCU 6 Type Error | 82 |
| SLC 03 Trouble | 58 | OCU 7 Type Error | 83 |
| SLC 04 Trouble | 59 | OCU 8 Type Error | 84 |
| SLC 05 Trouble | 60 | SWM Trouble | 85 |
| SLC 06 Trouble | 61 | Connection Error | 86 |
| SLC 07 Trouble | 62 | Remote Annunciator #01 Trouble | 87 |
| SLC 08 Trouble | 63 | Remote Annunciator #02 Trouble | 88 |
| SLC 09 Trouble | 64 | Remote Annunciator #03 Trouble | 89 |
| SLC 10 Trouble | 65 | Remote Annunciator #04 Trouble | 90 |
| SLC 11 Trouble | 66 | Remote Annunciator #05 Trouble | 91 |
| SLC 12 Trouble | 67 | Remote Annunciator #06 Trouble | 92 |
| NIU Board Trouble | 68 | Remote Annunciator #07 Trouble | 93 |
| OCU 1 Trouble | 69 | Remote Annunciator #08 Trouble | 94 |
| OCU 2 Trouble | 70 | Remote Annunciator #09 Trouble | 95 |
| OCU 3 Trouble | 71 | Remote Annunciator #10 Trouble | 96 |
| OCU 4 Trouble | 72 | Remote Annunciator #11 Trouble | 97 |
| OCU 5 Trouble | 73 | Remote Annunciator #12 Trouble | 98 |
| OCU 6 Trouble | 74 | Remote Annunciator #13 Trouble | 99 |
| OCU 7 Trouble | 75 | Remote Annunciator #14 Trouble | 100 |

Table of System/Panel Troubles and offset

| System Trouble Event | Data Array Offset | System Trouble Event | Data Array Offset |
|---|---|---|---|
| Remote Annunciator #15 Trouble | 101 | Voice IC Power Trouble | 126 |
| Remote Annunciator #16 Trouble | 102 | RTC Trouble | 127 |
| Remote Annunciator #17 Trouble | 103 | Fuse Trouble | 128 |
| Remote Annunciator #18 Trouble | 104 | Speaker Trouble | 129 |
| Remote Annunciator #19 Trouble | 105 | Telephone jack Trouble | 130 |
| Remote Annunciator #20 Trouble | 106 | Trouble Input On | 131 |
| Remote Annunciator #21 Trouble | 107 | Fire Drill | 132 |
| Remote Annunciator #22 Trouble | 108 | Signal  Silence | 133 |
| Remote Annunciator #23 Trouble | 109 | General Alarm | 134 |
| Remote Annunciator #24 Trouble | 110 | Configuration Data Download | 135 |
| Remote Annunciator #25 Trouble | 111 | Configuration Data Upload | 136 |
| Remote Annunciator #26 Trouble | 112 | Program Data Download | 137 |
| Remote Annunciator #27 Trouble | 113 | Event Log Data Upload | 138 |
| Remote Annunciator #28 Trouble | 114 | Maintenance List Data Upload | 139 |
| Remote Annunciator #29 Trouble | 115 | Network Ground Fault | 140 |
| Remote Annunciator #30 Trouble | 116 | Network Failure Port A | 141 |
| RA Loop-Back | 117 | Network Failure Port B | 142 |
| CTM Trouble | 118 | Duplicate Node Numbers | 143 |
| DACT Trouble | 119 | Node #01 Network Failure | 144 |
| HOST Trouble | 120 | Node #02 Network Failure | 145 |
| BMS Trouble | 121 | Node #03 Network Failure | 146 |
| PRN Trouble | 122 | Node #04 Network Failure | 147 |
| LCD Trouble | 123 | Node #05 Network Failure | 148 |
| CPU Trouble | 124 | Node #06 Network Failure | 149 |
| Voice IC Trouble | 125 | Node #07 Network Failure | 150 |

Table of System/Panel Troubles and offset

| System Trouble Event | Data Array Offset | System Trouble Event | Data Array Offset |
|---|---|---|---|
| Node #08 Network Failure | 151 | Node #43 Network Failure | 176 |
| Node #09 Network Failure | 152 | Node #44 Network Failure | 177 |
| Node #10 Network Failure | 153 | Node #45 Network Failure | 178 |
| Node #11 Network Failure | 154 | Node #46 Network Failure | 179 |
| Node #12 Network Failure | 155 | Node #47 Network Failure | 180 |
| Node #13 Network Failure | 156 | Node #48 Network Failure | 181 |
| Node #14 Network Failure | 157 | Node #49 Network Failure | 182 |
| Node #15 Network Failure | 158 | Node #50 Network Failure | 183 |
| Node #16 Network Failure | 159 | Node #51 Network Failure | 184 |
| Node #17 Network Failure | 160 | Node #52 Network Failure | 185 |
| Node #18 Network Failure | 161 | Node #53 Network Failure | 186 |
| Node #19 Network Failure | 162 | Node #54 Network Failure | 187 |
| Node #20 Network Failure | 163 | Node #55 Network Failure | 188 |
| Node #21 Network Failure | 164 | Node #56 Network Failure | 189 |
| Node #22 Network Failure | 165 | Node #57 Network Failure | 190 |
| Node #23 Network Failure | 166 | Node #58 Network Failure | 191 |
| Node #24 Network Failure | 167 | Node #59 Network Failure | 192 |
| Node #25 Network Failure | 168 | Node #60 Network Failure | 193 |
| Node #26 Network Failure | 169 | Node #61 Network Failure | 194 |
| Node #27 Network Failure | 170 | Node #62 Network Failure | 195 |
| Node #28 Network Failure | 171 | Node #63 Network Failure | 196 |
| Node #29 Network Failure | 172 | Node #64 Network Failure | 197 |
| Node #30 Network Failure | 173 | Ground Fault | 198 |
| Node #41 Network Failure | 174 | | |
| Node #42 Network Failure | 175 | | |

### 4.5.3   Device/Module Events

A device event, for the purposes of this manual, is one reported by an addressable device on a loop.

The driver is configured by making a map descriptor for each loop on each panel. For each type of event you are interested in you make a map descriptor. And again, for each circuit that is used you need another set of Map Descriptors. The offset into the data array is the device number.

Eg. If you are using a Data Array called **DA_N1L1_Alert** to store Alert events for Loop 1 on panel 1, then when device 1 reports an alert the driver will set DA_N1L1_Alert[1]=1. If device 10 reports an alert event, then DA_N1L1_Alert[10]=1. When the event clears then the value in the Data Array is set to zero.

Eg. If you are using a Data Array called **DA_N1L1_Alarms** to store Alarm events for Loop 1 on panel 1, then when device 1 report an Alarm the driver will set **DA_N1L1_Alarms** [1]=1. If device 10 reports an Alarm event then **DA_N1L1_Alarms** [10]=1. When the event clears then the value in the Data Array is set to zero.

### 4.5.4   Summarizing Data

A single device on a single loop can in be in multiple states simultaneously. For example, a Smoke detector can report an *alert*, an *action* or an *alarm* event. Normally these events are stored in different locations / Data Arrays.

**What happens if you don't care if its in alarm or alert, what you care about is the abnormal state?** Then you should use the Summary system.

To do this,  allocate a Data Array to store the summary. The Array Data type should be UINT16 – a 16 bit word. The driver allocates a bit for each state. Thus if a device is in multiple states then multiple bits will be set. You simply check the word to see if it is zero (no bits set meaning all normal) or non-zero meaning at least one active state. You could go further and extract the bits to learn the specific state. (See the FieldServer Bridge Configuration Manual for notes on how to use the Bit_Extract function.)

This system allocates separate summaries for actives and troubles so you need to use two different data arrays.

#### 4.5.4.1   Summary

If you specify the parameter **Multicrest_Summary_DA** then when the driver stores the event it will also update this summary Data Array. Again, the Data Array offset number is equal to the device number. In this case, the driver does more than store a 1 or a zero. It set the bit in the number to 1 or zero. Thus the value changes as the events go on/off. The following table shows the relationship between bit number and event type.

Note how some bit numbers are used twice. That means you cannot use this summary method to summarize troubles and actives together. Using this method you should use one Data Array for Actives and a different one for Troubles (an example is provided below) The Data arrays need have type=UINT16 so there are enough bits available for setting.

| ACTIVATION EVENTS | | | TROUBLE EVENTS | |
|---|---|---|---|---|
| EVENT TYPE | BIT NUMBER | | EVENT TYPE | BIT NUMBER |
| Alert | 0 | | Device Fault | 0 |
| Action | 1 | | Device Mismatch | 1 |

| | | | |
|---|---|---|---|
| Confirm | 2 | Device Bypass | 2 |
| Alarm | 3 | Missing Device | 3 |
| AlarmF1 | 4 | Multiple Address | 4 |
| AlarmF2 | 5 | Low Temp | 5 |
| Verify | 6 | Sensor Fault | 6 |
| Supervisory | 7 | Device Dirty | 7 |
| Active | 8 | Fault | 8 |
| DischargeCnf | 9 | | |
| PreDischarge | 10 | | |
| Discharge | 11 | | |
| | | | |
| | | | |

While it may make sense to use Data Arrays whose format=BIT for the events, for the summary Data Array it makes sense to use a UINT16 .

Example:

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr     ,Multicrest_Summary_DA

StoreClassific      ,N1_LOOP1_class  ,0                 ,Server   ,Node1     ,Classification      ,1:1                 ,-                ,

StoreAlerts         ,N1_LOOP1        ,0                 ,Server   ,Node1     ,Any_Alarm           ,1:1                 ,N1_LOOP1_ACTIVES ,

StoreNo_Asnwer      ,N1_LOOP1        ,0                 ,Server   ,Node1     ,Any_Trble           ,1:1                 ,N1_LOOP1_TRBLS   ,
```

### 4.5.5   MultiState Data

The driver presents a system for building a MultiState value for device / module states.

If a device is in an alarm state, the driver provides a number whose value reports the most important / serious of the possible active states. For example,  if a device is in Alert and Alarm States simultaneously then the driver will report the Alarm state since it is considered a higher priority.

If a device is a trouble state, then the driver provides a number whose value reports which trouble state is active (if only one state is active) or which reports that multiple states are active when more than one trouble is active.

### 4.5.5.1  MultiState Alarms

The driver can set a Data Array location to one of the tabulated values below based on the current activated *Alarm* states.  If any two or more states are active at the same time, the driver sets the value to the one with the highest value. Eg. If Alert and Alarm States for a device are active then the Multistate Value will be set to 4 because Alarm (4) has a higher value than Alert (1).

| MultiState Description | Value |
| --- | --- |
| All Normal | 1 |
| Alert | 2 |
| Action | 3 |
| Confirm | 4 |
| Alarm | 5 |
| AlarmF1 | 6 |
| AlarmF2 | 7 |
| Verify | 8 |
| Supervisory | 9 |
| Active | 10 |
| DischargeCnf | 11 |
| PreDischarge | 12 |
| DischargeX | 13 |
|  |  |

### 4.5.5.2  MultiState Troubles

The driver can set a Data Array location to one of the tabulated values below based on the current activated Trouble states.  If any two or more states are active at the same time, the driver sets the value to indicate that there are multiple troubles present.

| MultiState Description | Value |
| --- | --- |
| No Troubles Present | 1* |
| Device Fault | 2 |
| Device Mismatch | 3 |
| Device Bypass | 4 |
| Missing Device | 5 |
| Multiple Address | 6 |
| Low Temp | 7 |
| Sensor Fault | 8 |
| Device Dirty | 9 |
| Fault | 10 |
| Multiple Troubles present | 15 |

This system requires that you implement the summary described in section 4.6.4 **Error! Reference source not found.** or it will not operate correctly. See section for an example.

*The value zero might be used when the gateway starts up. This depends on configuration. A preload of 1 will resolve this.

### 4.5.6   On Reset

Device and summary data is cleared when there is a System Reset. Normally all Multicrest data is cleared irrespective of node but this can be controlled – See Statistic #50 in the appendix.

## 4.6    Some More Example

### 4.6.1    Map Descriptor Example 1 – Store System Troubles

System Troubles are those reported by the Main MCU. Troubles reported by networked panels/nodes are called Panel Troubles by this driver. Therefore you can only have one Map Descriptor capturing System Troubles but you can have multiple Map Descriptors capturing Panel Troubles.  If a System Trouble Event is reported by 'Node1' then the Data Array N1_SysTrb will be updated. The offset into the data array indicates the type of system trouble.  It is important that in the Node section of the config, that the main panel is allocated a Node_ID which corresponds to the Main Panel number and that the Node Name is correctly spelled.

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name  ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr        ,Multicrest_Summary_DA

StoreSysTroubles    ,N1_SYSTRB        ,0                 ,Server   ,Node1     ,System_Trouble   ,-                     ,-                  ,
```

| Descriptive only. Not used. | Data Array and offset where data will be stored. | Wait passively for events to be reported. | Node Name of a Node defined in the Nodes section. Node Number must correspond to the Multicrest panel number. Name must match too. | Store System Troubles (for this node) | Not required when storing System Troubles | Not required when storing System Troubles |
|---|---|---|---|---|---|---|

### 4.6.2 Map Descriptor Example 2 – Store Panel Troubles

System Troubles are those reported by the Main MCU. Troubles reported by networked panels/nodes are called Panel Troubles by this driver. You can have multiple Map Descriptors capturing Panel Troubles.  If a Panel Trouble Event is reported by 'Node2' then the Data Array N2_SysTrb will be updated. The offset into the data array indicates the type of system trouble.  It is important that in the Node section of the config, that the main panel is allocated a Node_ID which corresponds to the Networked  Panel number **and that the Node Name is correctly spelled**.

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name  ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr        ,Multicrest_Summary_DA

StoreSysTroubles    ,N1_SYSTRB        ,0                 ,Server   ,Node1     ,Panel_Trouble     ,-                        ,-                 ,
```

| Descriptive only. Not used. | Data Array and offset where data will be stored. | Wait passively for events to be reported. | Node Name of a Node defined in the Nodes section. Node Number must correspond to the Multicrest panel number. Name must match too. | Store System Troubles (for this node) | Not required when storing System Troubles | Not required when storing System Troubles |

### 4.6.3   Map Descriptor Example 3 – Store Classifications

You can have the driver store a number corresponding to the type or classification of an addressable point on a loop (eg a device) using one Map Descriptor per loop per panel.

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name  ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr        ,Multicrest_Summary_DA

StoreClassific      ,N1_LOOP1_class   ,0                 ,Server   ,Node1     ,Classification      ,1:1                    ,-                    ,
```

| Descriptive only. Not used. | Data Array and offset where data will be stored. | Wait passively for events to be reported. | Node Name of a Node defined in the Nodes section. Node Number must correspond to the Multicrest panel number. Name must match too. | Store 'Classification' Data for this loop on this node. | Specify which loop and circuit  Separated by a colon. | Not required when storing System Troubles |
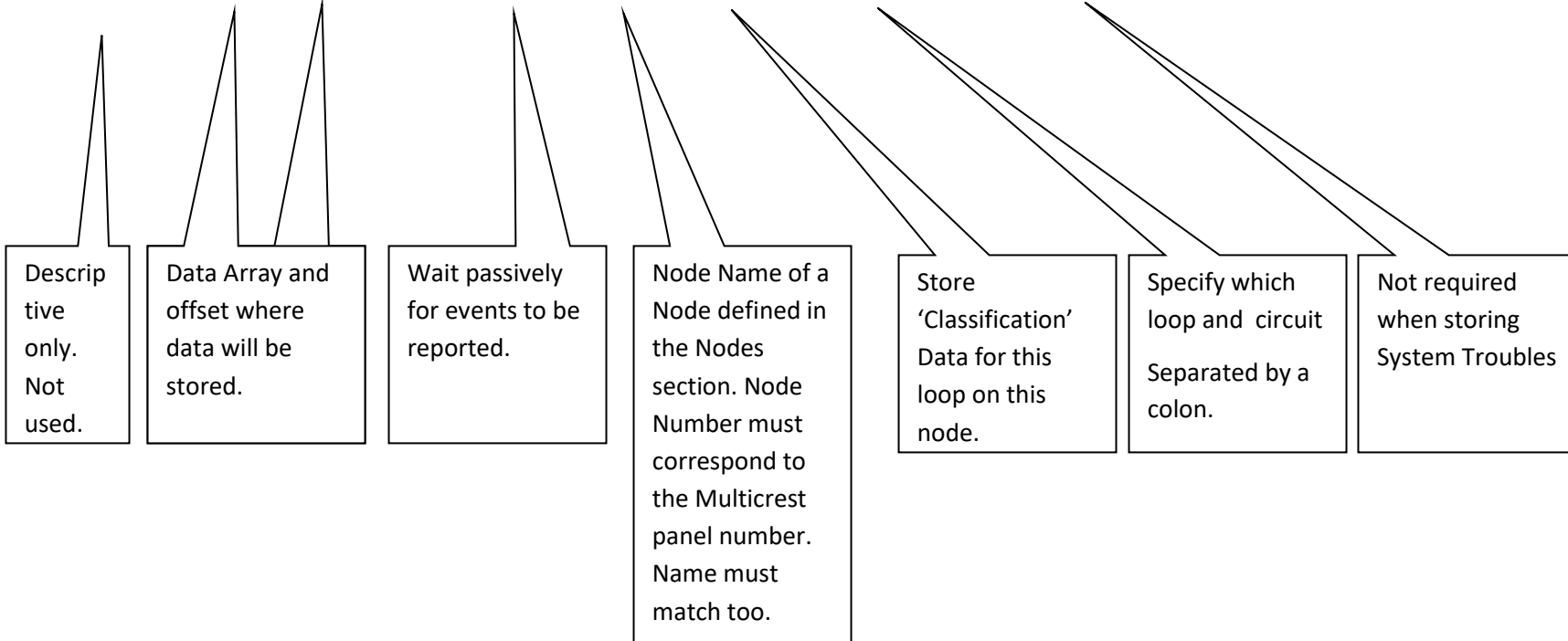
### 4.6.4 Map Descriptor Example 4 – Store Device Event Data

In this example Alert Events for Loop 2 on Node 1 are processed. Each time one of these events reports the driver looks for a Map Descriptor whose Multicrest_Function=Alert and whose Loop Number and node match. If one is found, the driver looks up the specified Data Array and offset and sets the value to 1 (On) or zero(off). The offset number equals the device number (device's address on the loop). So if Loop 2 Device 10 reports an Alert On then expect Data Array N1_LOOP2_ALERT[10] to be 1.

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name  ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr      ,Multicrest_Summary_DA

StoreAlerts          ,N1_LOOP2_ALERT   ,0                 ,Server   ,Node1     ,Alert               ,2:1                  ,N1_LOOP2_SUMM   ,
```
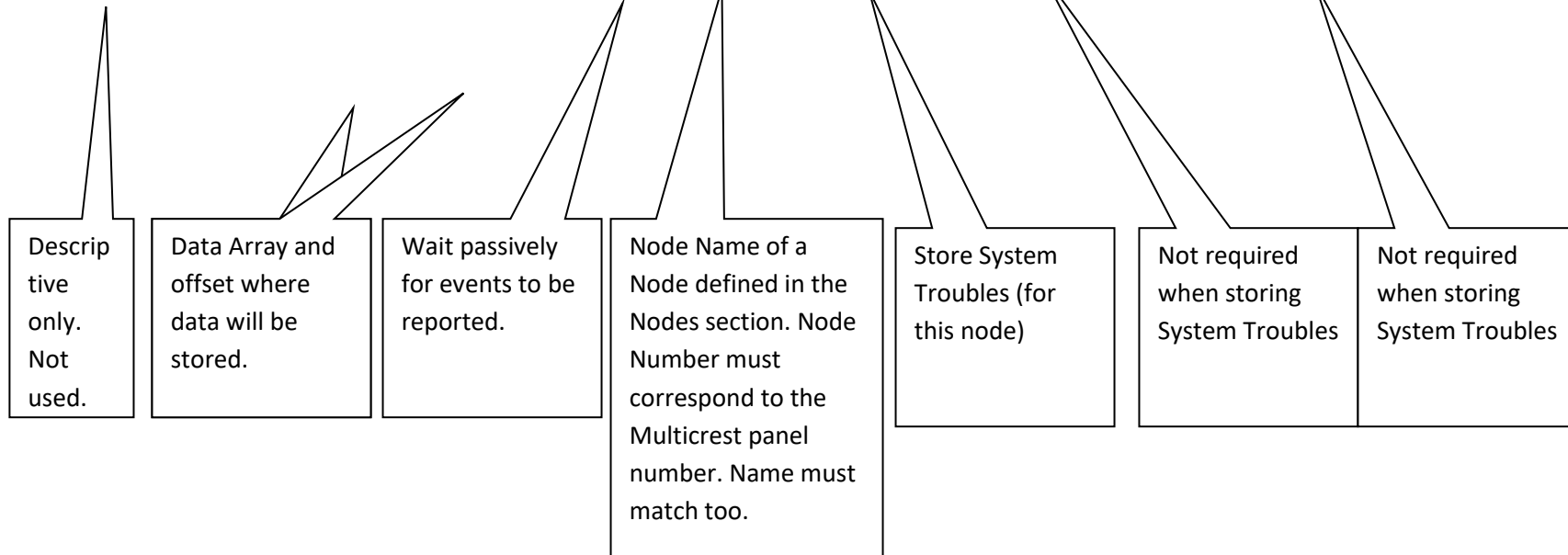
## 4.6.5 Map Descriptor Example 5 – Store Device Event Data

Like the previous example but this time, we have defined one map descriptor for each event type.

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name  ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr  ,Multicrest_Summary_DA

StoreAlerts         ,N1_LOOP2_ALERT   ,0                 ,Server   ,Node1     ,Alert               ,2:1              ,N1_LOOP2_SUMM  ,

StoreActions        ,N1_LOOP2_ACTION  ,0                 ,Server   ,Node1     ,Action              ,2:1              ,N1_LOOP2_SUMM  ,

StoreConfirms       ,N1_LOOP2_CONFIR  ,0                 ,Server   ,Node1     ,Confirm             ,2:1              ,N1_LOOP2_SUMM  ,

StoreAlarms         ,N1_LOOP2_ALARM   ,0                 ,Server   ,Node1     ,Alarm               ,2:1              ,N1_LOOP2_SUMM  ,

StoreVerify         ,N1_LOOP2_VERIFY  ,0                 ,Server   ,Node1     ,Verify              ,2:1              ,N1_LOOP2_SUMM  ,

StoreSupervisory    ,N1_LOOP2_SUPER   ,0                 ,Server   ,Node1     ,Supervisory         ,2:1              ,N1_LOOP2_SUMM  ,

StoreNo_Asnwer      ,N1_LOOP2_NO_ANS  ,0                 ,Server   ,Node1     ,No_Answer           ,2:1              ,N1_LOOP2_SUMM  ,

StoreCommFault      ,N1_LOOP2_COMMFL  ,0                 ,Server   ,Node1     ,Comm_Fault          ,2:1              ,N1_LOOP2_SUMM  ,

StoreDisable        ,N1_LOOP2_DISABL  ,0                 ,Server   ,Node1     ,Disable             ,2:1              ,N1_LOOP2_SUMM  ,

StoreTypeErrors     ,N1_LOOP2_TYPERR  ,0                 ,Server   ,Node1     ,Type_Error          ,2:1              ,N1_LOOP2_SUMM  ,

StoreFaults         ,N1_LOOP2_FAULT   ,0                 ,Server   ,Node1     ,Fault               ,2:1              ,N1_LOOP2_SUMM  ,

StoreDataErrors     ,N1_LOOP2_DATAER  ,0                 ,Server   ,Node1     ,Data_Error          ,2:1              ,N1_LOOP2_SUMM  ,
```

## 4.6.6 Map Descriptor Example 6 – Summarizing Data

In this example 2 different summary Data Arrays are specified. One summarizes activation states and the other summarizes trouble states.  For example: If an ALARM event occurs on N1-1002  then the driver updates N1_LOOP1_ALERT [2] by setting its value to 1. It also goes to the Data Array N1_LOOP1_ALARM[2] and sets bit 3 in the N1_LOOP1_ACTIVES[2]. Offset 2 applies because the device address is 002. Bit 3 was modified in the existing value found N1_LOOP1_ACTIVES[2] because when we read section **Error! Reference source not found.  Error! Reference source not found.** we learn that Alarms affect Bit 3. So if the previous value in N1_LOOP1_ACTIVES[2] was zero then after the Alarm the value will be 8 (bit 3 set).

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name  ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr        ,Multicrest_Summary_DA

StoreClassific      ,N1_LOOP1_class   ,0                 ,Server   ,Node1     ,Classification      ,1:1                    ,-                    ,

StoreAlerts         ,N1_LOOP1_ALERT   ,0                 ,Server   ,Node1     ,Alert               ,1:1                    ,N1_LOOP1_ACTIVES   ,

StoreActions        ,N1_LOOP1_ACTION  ,0                 ,Server   ,Node1     ,Action              ,1:1                    ,N1_LOOP1_ACTIVES   ,

StoreConfirms       ,N1_LOOP1_CONFIR  ,0                 ,Server   ,Node1     ,Confirm             ,1:1                    ,N1_LOOP1_ACTIVES   ,

StoreAlarms         ,N1_LOOP1_ALARM   ,0                 ,Server   ,Node1     ,Alarm               ,1:1                    ,N1_LOOP1_ACTIVES   ,

StoreVerify         ,N1_LOOP1_VERIFY  ,0                 ,Server   ,Node1     ,Verify              ,1:1                    ,N1_LOOP1_ACTIVES   ,

StoreSupervisory    ,N1_LOOP1_SUPER   ,0                 ,Server   ,Node1     ,Supervisory         ,1:1                    ,N1_LOOP1_ACTIVES   ,

StoreNo_Asnwer      ,N1_LOOP1_NO_ANS  ,0                 ,Server   ,Node1     ,No_Answer           ,1:1                    ,N1_LOOP1_TRBLS     ,

StoreCommFault      ,N1_LOOP1_COMMFL  ,0                 ,Server   ,Node1     ,Comm_Fault          ,1:1                    ,N1_LOOP1_TRBLS     ,

StoreDisable        ,N1_LOOP1_DISABL  ,0                 ,Server   ,Node1     ,Disable             ,1:1                    ,N1_LOOP1_TRBLS     ,

StoreTypeErrors     ,N1_LOOP1_TYPERR  ,0                 ,Server   ,Node1     ,Type_Error          ,1:1                    ,N1_LOOP1_TRBLS     ,

StoreFaults         ,N1_LOOP1_FAULT   ,0                 ,Server   ,Node1     ,Fault               ,1:1                    ,N1_LOOP1_TRBLS     ,

StoreDataErrors     ,N1_LOOP1_DATAER  ,0                 ,Server   ,Node1     ,Data_Error          ,1:1                    ,N1_LOOP1_TRBLS     ,
```

### 4.6.7    Map Descriptor Example 7  –  MultiState Data

This example uses example 6 and expands on it.  The Data Array N1_L1_MultiALM  will contain a number representing the Alarm state as a MultiState value. The offset into the Data Array is equal to the device/module reporting the events. The Data Array N1_L1_MultiTRB  will contain a number reporting the single active trouble event or that multiple events are active at the same time.

Notes on the next page provide some information on how the value would change for a series of events.

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name ,Data_Array , ... ,Multicrest_Function ,Multicrest_Addr ,Multicrest_Summary ,Multicrest_MultiTrbls,Multicrest_MultiAlarm
                    ,                ,_Offset    ,     ,                    ,                ,_DA               ,_DA                  ,_DA
StoreAlerts         ,N1_LOOP1_ALERT  ,0          , ... ,Alert               ,1:1            ,N1_LOOP1_ACTIVE   ,-                    ,N1_L1_MultiALM
StoreActions        ,N1_LOOP1_ACTION ,0          , ... ,Action              ,1:1            ,N1_LOOP1_ACTIVE   ,-                    ,N1_L1_MultiALM
StoreConfirms       ,N1_LOOP1_CONFIR ,0          , ... ,Confirm             ,1:1            ,N1_LOOP1_ACTIVE   ,-                    ,N1_L1_MultiALM
StoreAlarms         ,N1_LOOP1_ALARM  ,0          , ... ,Alarm               ,1:1            ,N1_LOOP1_ACTIVE   ,-                    ,N1_L1_MultiALM
StoreVerify         ,N1_LOOP1_VERIFY ,0          , ... ,Verify              ,1:1            ,N1_LOOP1_ACTIVE   ,-                    ,N1_L1_MultiALM
StoreSupervisory    ,N1_LOOP1_SUPER  ,0          , ... ,Supervisory         ,1:1            ,N1_LOOP1_ACTIVE   ,-                    ,N1_L1_MultiALM
StoreNo_Asnwer      ,N1_LOOP1_NO_ANS ,0          , ... ,No_Answer           ,1:1            ,N1_LOOP1_TRBLS    ,N1_L1_MultiTRB       ,-
StoreCommFault      ,N1_LOOP1_COMMFL ,0          , ... ,Comm_Fault          ,1:1            ,N1_LOOP1_TRBLS    ,N1_L1_MultiTRB       ,-
StoreDisable        ,N1_LOOP1_DISABL ,0          , ... ,Disable             ,1:1            ,N1_LOOP1_TRBLS    ,N1_L1_MultiTRB       ,-
StoreTypeErrors     ,N1_LOOP1_TYPERR ,0          , ... ,Type_Error          ,1:1            ,N1_LOOP1_TRBLS    ,N1_L1_MultiTRB       ,-
StoreFaults         ,N1_LOOP1_FAULT  ,0          , ... ,Fault               ,1:1            ,N1_LOOP1_TRBLS    ,N1_L1_MultiTRB       ,-
StoreDataErrors     ,N1_LOOP1_DATAER ,0          , ... ,Data_Error          ,1:1            ,N1_LOOP1_TRBLS    ,N1_L1_MultiTRB       ,-
```

Assume all these events occur for a device at N01-1002. Then the Data Array offset affected is offset 2 (corresponds to device address).

Assume that a system reset was performed and that there are no active events before the ones listed below.

ALERT on  event.; Bit 0: Value of $_{N1\_LOOP1\_ACTIVE}$[2]  before event  = 0    Value after update = 1 (Binary=1)

        N1_L1_MultiALM[2] = 1

ACTION  event.; Bit1 : Value of $_{N1\_LOOP1\_ACTIVE}$[2]  before event  = 1    Value after update = 3 (Binary=11)

        N1_L1_MultiALM[2] = 2 (Action(2) is more important than Alert(1))

ALARM on  event.; Bit 3: Value of $_{N1\_LOOP1\_ACTIVE}$[2]  before event  = 3    Value after update = 11 (Binary=1011)

        N1_L1_MultiALM[2] = 4 (Alarm(4) is more important than Action(2) which is more important than Alert(1))


NO ANSWER  on event.; Bit 0: Value of $_{N1\_LOOP1\_TRBLS}$[2]  before event  =0    Value after update = 1 (Binary= 1)

        N1_L1_MultiTRB[2] = 1 (No Asnwer) (See table in section **Error! Reference source not found. Error! Reference source not found.** )

FAULT  on event.; Bit 4: Value of $_{N1\_LOOP1\_TRBLS}$[2]  before event  =1    Value after update = 17 (Binary= 10001)

        N1_L1_MultiTRB[2] = 15 (Multiple Trouble active)

NO ANSWER  **off** event.; Bit 0: Value of $_{N1\_LOOP1\_TRBLS}$[2]  before event  =17    Value after update =161 (Binary= 10000)

        N1_L1_MultiTRB[2] = 5 (Fault) (See table in section **Error! Reference source not found. Error! Reference source not found.** )

### 4.6.8 Map Descriptor Example 8 – Simplified Config for MultiState Data

In this example a We have used the functions 'Any_Alarm' and 'Any_Trbl' . When this is done some information is lost – the driver does not store the individual state bits for each device. Rather, it simply maintains the summary and multistate. When an event occurs the driver decides if it's an activation event or a trouble event. It looks for a place to store. If it's an activation it looks for a MD whose function is specific to the event  and/or it looks for a MD whose function=Any…. Alarm/Trbl and uses the summary and multistate arrays specified.

```
Map_Descriptors

Map_Descriptor_Name ,Data_Array_Name  ,Data_Array_Offset ,Function ,Node_Name ,Multicrest_Function ,Multicrest_Addr ,Multicrest_Summary_DA ,Multicrest_MultiTrbls_DA,Multicrest_MultiAlarm_DA

StoreActivations    ,N1_DUMMY         ,0                 ,Server   ,Node1     ,Any_alarm           ,1:1             ,N1_LOOP1_ACTIVE       ,-                       ,N1_L1_MultiALM

StoreTroubles       ,N1_DUMMY         ,0                 ,Server   ,Node1     ,Any_trbl            ,1:1             ,N1_LOOP1_TRBLS        ,N1_L1_MultiTRB          ,-
```

# 5    Configuring the FieldServer as a Multicrest FACP Server

This driver cannot be used to emulate an Multicrest Panel.

## 6   Revision History

| DATE | RESP | FORMAT | DRIVER VER. | DOC. REV. | COMMENT |
|------|------|--------|-------------|-----------|---------|
| 20 Aug 2014 | PMC | | 1.0 | 1 | Created |
| 28 Aug 2014 | GFM | | *1.0* | 2 | *QC* |
| 09 Sep 2014 | PMC | | 1.0 | 3 | Updated |
| 29 Sep 2014 | PMC | | 1.0 | 4 | Appendix A.3 Updated. |
| 01 Oct 2014 | PMC | | 1.01 | 5 | Multistate value and table updated. |
| 10 May 2021 | YC | | 1.01 | 6 | Updated to latest template |

# Appendix A. Advanced Topics

## Appendix A.1 Driver Error Messages

A consequence of the fact that this is a passive client driver, is that the FieldServer must be synchronized with the panel by resetting the panel and then restarting the FieldServer to reset the data.

Ensure all points are normal and there are no active conditions. Perform a system Reset. This will result in a synch between panel and gateway. Even if you do not clear all active conditions 1$^{st}$ this method may be effective since, the panel should re-send status messages reporting all active conditions after the reset. You should test to ensure this method works for you before using it.

Read more – see the section on Driver Statistics and read stat #50 notes.

## Appendix A.2 Loading New Classifications

A driver contains a defualt list of classfications. The list may be updated. The driver looks for and if found, loads a file called classi.ini. In this case it prints a message and lsists all the classifications that were loaded. When classi.ini is loaded the default table is ignored. A maximum of 100 can be loaded.

| Classifcation | Value | Classifcation | Value |
|---|---|---|---|
| Analog Smk Detector | 1 | Silent Input | 35 |
| Fixed Heat Detector | 2 | Auto Mode | 36 |
| Comb. Heat Detector | 3 | Manual Mode | 37 |
| Smk/Fixed Detector | 4 | Action Input | 38 |
| DUCT Detector | 5 | Alert Input | 39 |
| Photo Smk Detector | 6 | Control Output | 40 |
| Ion Smoke Detector | 7 | Bell Circuit | 41 |
| Heat Detector | 8 | Horn Circuit | 42 |
| Smoke/Heat Detector | 9 | Strobe Circuit | 43 |
| Leak Detector | 10 | Audible Circuit | 44 |
| Pull Station | 11 | Speaker Circuit | 45 |
| Monitor Module | 12 | Release Bell | 46 |
| Beam Smk Detector | 13 | Discharge Confirm | 47 |
| Gas Module | 14 | Telephone | 48 |
| Waterflow | 15 | Relay | 49 |
| Manual Release | 16 | Power Shutdown | 50 |
| Second Shot | 17 | HVAC Shutdown | 51 |

| | | | |
|---|---|---|---|
| Sprinkler System | 18 | FAN Shutdown | 52 |
| Monitor Point | 19 | Release Form C | 53 |
| Alarm Input | 20 | Fire Alarm Status | 54 |
| Supervisory | 21 | PreDischarge Status | 55 |
| Tamper | 22 | Supervisory Status | 56 |
| Abort Switch | 23 | Gas Leak Status | 57 |
| Sup Fire Pmp Run | 24 | Trouble Status | 58 |
| Sup Fire Pmp Flt | 25 | AC Fault Status | 59 |
| Trouble Input | 26 | BATT Fault Status | 60 |
| Fault Input | 27 | BYpass Status | 61 |
| Leak Detector Fault | 28 | System Reset Status | 62 |
| Signal Silence | 29 | Silence Status | 63 |
| Fire Drill | 30 | Drill Status | 64 |
| System Reset | 31 | Waterflow Status | 65 |
| General Alarm | 32 | Analog Sounder Base | 66 |
| PAS Inhibit | 33 | Analog Relay Base | 67 |
| Hazard Alert | 34 | Release Circuit | 68 |

## Appendix A.3   Loading New System/Panel Troubles Event Types

The driver contains a list of default trouble strings. This list can be updated. . The driver looks for and if found, loads a file called systrb.ini. In this case it prints a message and lists all the trouble strings that were loaded. When systrb.ini is loaded the default table is ignored. A maximum of 200 System Trouble Event types can be loaded.

Default Values are tabulated in section **Error! Reference source not found. Error! Reference source not found.**:-

| System/Panel Trouble Type | Enumeration | System/Panel Trouble Type | Enumeration |
| --- | --- | --- | --- |
| Configuration Data Error | 1 | SLC 07 Open | 26 |
| PSU 1 AC Power Fault | 2 | SLC 08 Open | 27 |
| PSU 2 AC Power Fault | 3 | SLC 09 Open | 28 |
| PSU 1 Battery Fault | 4 | SLC 10 Open | 29 |
| PSU 2 Battery Fault | 5 | SLC 11 Open | 30 |
| PSU 1 Ground Fault | 6 | SLC 12 Open | 31 |
| PSU 2 Ground Fault | 7 | SLC 01 Short | 32 |
| SCU 1 Ground Fault | 8 | SLC 02 Short | 33 |
| SCU 2 Ground Fault | 9 | SLC 03 Short | 34 |
| SCU 3 Ground Fault | 10 | SLC 04 Short | 35 |
| SCU 4 Ground Fault | 11 | SLC 05 Short | 36 |
| SCU 5 Ground Fault | 12 | SLC 06 Short | 37 |
| SCU 6 Ground Fault | 13 | SLC 07 Short | 38 |
| PSU 1 24V Fault | 14 | SLC 08 Short | 39 |
| PSU 2 24V Fault | 15 | SLC 09 Short | 40 |
| PSU 1 Charging Ciruit Fault | 16 | SLC 10 Short | 41 |
| PSU 2 Charging Ciruit Fault | 17 | SLC 11 Short | 42 |
| PSU 1 Board Trouble | 18 | SLC 12 Short | 43 |
| PSU 2 Board Trouble | 19 | SLC 01 Power Fault | 44 |
| SLC 01 Open | 20 | SLC 02 Power Fault | 45 |
| SLC 02 Open | 21 | SLC 03 Power Fault | 46 |
| SLC 03 Open | 22 | SLC 04 Power Fault | 47 |
| SLC 04 Open | 23 | SLC 05 Power Fault | 48 |
| SLC 05 Open | 24 | SLC 06 Power Fault | 49 |
| SLC 06 Open | 25 | SLC 07 Power Fault | 50 |

Default System/Panel Troubles continued:

| System/Panel Trouble Type | Enumeration | System/Panel Trouble Type | Enumeration |
|---|---|---|---|
| SLC 08 Power Fault | 51 | OCU 8 Trouble | 76 |
| SLC 09 Power Fault | 52 | OCU 1 Type Error | 77 |
| SLC 10 Power Fault | 53 | OCU 2 Type Error | 78 |
| SLC 11 Power Fault | 54 | OCU 3 Type Error | 79 |
| SLC 12 Power Fault | 55 | OCU 4 Type Error | 80 |
| SLC 01 Trouble | 56 | OCU 5 Type Error | 81 |
| SLC 02 Trouble | 57 | OCU 6 Type Error | 82 |
| SLC 03 Trouble | 58 | OCU 7 Type Error | 83 |
| SLC 04 Trouble | 59 | OCU 8 Type Error | 84 |
| SLC 05 Trouble | 60 | SWM Trouble | 85 |
| SLC 06 Trouble | 61 | Connection Error | 86 |
| SLC 07 Trouble | 62 | Remote Annunciator #01 Trouble | 87 |
| SLC 08 Trouble | 63 | Remote Annunciator #02 Trouble | 88 |
| SLC 09 Trouble | 64 | Remote Annunciator #03 Trouble | 89 |
| SLC 10 Trouble | 65 | Remote Annunciator #04 Trouble | 90 |
| SLC 11 Trouble | 66 | Remote Annunciator #05 Trouble | 91 |
| SLC 12 Trouble | 67 | Remote Annunciator #06 Trouble | 92 |
| NIU Board Trouble | 68 | Remote Annunciator #07 Trouble | 93 |
| OCU 1 Trouble | 69 | Remote Annunciator #08 Trouble | 94 |
| OCU 2 Trouble | 70 | Remote Annunciator #09 Trouble | 95 |
| OCU 3 Trouble | 71 | Remote Annunciator #10 Trouble | 96 |
| OCU 4 Trouble | 72 | Remote Annunciator #11 Trouble | 97 |
| OCU 5 Trouble | 73 | Remote Annunciator #12 Trouble | 98 |
| OCU 6 Trouble | 74 | Remote Annunciator #13 Trouble | 99 |
| OCU 7 Trouble | 75 | Remote Annunciator #14 Trouble | 100 |

Default System/Panel Troubles continued:

| System/Panel Trouble Type | Enumeration | System/Panel Trouble Type | Enumeration |
|---|---|---|---|
| Remote Annunciator #15 Trouble | 101 | Voice IC Power Trouble | 126 |
| Remote Annunciator #16 Trouble | 102 | RTC Trouble | 127 |
| Remote Annunciator #17 Trouble | 103 | Fuse Trouble | 128 |
| Remote Annunciator #18 Trouble | 104 | Speaker Trouble | 129 |
| Remote Annunciator #19 Trouble | 105 | Telephone jack Trouble | 130 |
| Remote Annunciator #20 Trouble | 106 | Trouble Input On | 131 |
| Remote Annunciator #21 Trouble | 107 | Fire Drill | 132 |
| Remote Annunciator #22 Trouble | 108 | Signal  Silence | 133 |
| Remote Annunciator #23 Trouble | 109 | General Alarm | 134 |
| Remote Annunciator #24 Trouble | 110 | Configuration Data Download | 135 |
| Remote Annunciator #25 Trouble | 111 | Configuration Data Upload | 136 |
| Remote Annunciator #26 Trouble | 112 | Program Data Download | 137 |
| Remote Annunciator #27 Trouble | 113 | Event Log Data Upload | 138 |
| Remote Annunciator #28 Trouble | 114 | Maintenance List Data Upload | 139 |
| Remote Annunciator #29 Trouble | 115 | Network Ground Fault | 140 |
| Remote Annunciator #30 Trouble | 116 | Network Failure Port A | 141 |
| RA Loop-Back | 117 | Network Failure Port B | 142 |
| CTM Trouble | 118 | Duplicate Node Numbers | 143 |
| DACT Trouble | 119 | Node #01 Network Failure | 144 |
| HOST Trouble | 120 | Node #02 Network Failure | 145 |
| BMS Trouble | 121 | Node #03 Network Failure | 146 |
| PRN Trouble | 122 | Node #04 Network Failure | 147 |
| LCD Trouble | 123 | Node #05 Network Failure | 148 |
| CPU Trouble | 124 | Node #06 Network Failure | 149 |
| Voice IC Trouble | 125 | Node #07 Network Failure | 150 |

Default System/Panel Troubles continued:

| System/Panel Trouble Type | Enumeration | System/Panel Trouble Type | Enumeration |
|---|---|---|---|
| Node #08 Network Failure | 151 | Node #43 Network Failure | 176 |
| Node #09 Network Failure | 152 | Node #44 Network Failure | 177 |
| Node #10 Network Failure | 153 | Node #45 Network Failure | 178 |
| Node #11 Network Failure | 154 | Node #46 Network Failure | 179 |
| Node #12 Network Failure | 155 | Node #47 Network Failure | 180 |
| Node #13 Network Failure | 156 | Node #48 Network Failure | 181 |
| Node #14 Network Failure | 157 | Node #49 Network Failure | 182 |
| Node #15 Network Failure | 158 | Node #50 Network Failure | 183 |
| Node #16 Network Failure | 159 | Node #51 Network Failure | 184 |
| Node #17 Network Failure | 160 | Node #52 Network Failure | 185 |
| Node #18 Network Failure | 161 | Node #53 Network Failure | 186 |
| Node #19 Network Failure | 162 | Node #54 Network Failure | 187 |
| Node #20 Network Failure | 163 | Node #55 Network Failure | 188 |
| Node #21 Network Failure | 164 | Node #56 Network Failure | 189 |
| Node #22 Network Failure | 165 | Node #57 Network Failure | 190 |
| Node #23 Network Failure | 166 | Node #58 Network Failure | 191 |
| Node #24 Network Failure | 167 | Node #59 Network Failure | 192 |
| Node #25 Network Failure | 168 | Node #60 Network Failure | 193 |
| Node #26 Network Failure | 169 | Node #61 Network Failure | 194 |
| Node #27 Network Failure | 170 | Node #62 Network Failure | 195 |
| Node #28 Network Failure | 171 | Node #63 Network Failure | 196 |
| Node #29 Network Failure | 172 | Node #64 Network Failure | 197 |
| Node #30 Network Failure | 173 | Ground Fault | 198 |
| Node #41 Network Failure | 174 | | End |
| Node #42 Network Failure | 175 | | |

## Appendix A.4    Unsupported Features

Any functionality not specifically specified as supported above can be regarded as excluded. In particular:


1) Panel configuration messages such as those detailed in sections 8 and 9 of the Multicrest specification are excluded.
2) Legacy messages such as those directed at the LaserCOMPACT device are excluded
3) Zone status' are excluded since there are no direct messages coming from the panel that indicate the status of the specific zones.
4) The Custom Message field (4th part of event message) is not used. If Zone info is printed in the customer message field then it is ignored as the whole field is always ignored.


## Appendix A.5    Zone Data

The driver does not process zone data reported in event messages.


## Appendix A.6    Simulation Data

The Multicrest Panel can be put in simulation mode and in this mode it reports events but marks them as simulated events. The driver does not differentiate between actual and simulated events.

## Appendix A.7    Driver Error Messages

| ERROR MESSAGE | EXPLANATION AND CORRECTIVE ACTION |
|---|---|
| We have shown place holders for the parts of the message which change. %s is a place holder for a text string. %d is a place holder for a number. %c is a place holder for an alpha character. | ***FYI messages are informational and do not require a corrective action. Simply use them to confirm configuration / behaviors are what you expect.***<br><br>Correct CSV* means  Update the configuration CSV file, download to gateway, restart gateway to give effect to the changes<br><br>Diagnostic Report^ means this error is unexpected. You should call for support. First try and capture diagnostic information. If your site is remote, call for help before you leave. |
| MCREST#01 Err Dont recognize <%s> | Multicrest_Function specified in config file is not recognized. Correct CSV* |
| MCREST#02 Err Dont recognize <%s> | Multicrest_Function specified in config file is not recognized. Correct CSV* |
| MCREST#03 Err Dont recognize Function#=<%d> | Diagnostic Report^ |
| MCREST#04 Err Dont recognize <%s> | Multicrest_Function specified in config file is not recognized. Correct CSV* |
| MCREST#05 Err. The Function Name must be specified. | The Map Descriptor parameter "Multicrest_Function" has not been specified. Correct CSV* |
| MCREST#10 Err Cant open <%s> | This message is produced when the firmware is tested. It should never be seen during normal operation. Diagnostic Report^ |
| MCREST#11 Err File <%s> doesnt have line=%d | This message is produced when the firmware is tested. It should never be seen during normal operation. Diagnostic Report^ |
| MCREST#12 Err File <%s> line=%d does not have 4 tokens | This message is produced when the firmware is tested. It should never be seen during normal operation. Diagnostic Report^ |
| MCREST#13 Err. Should not reach this code. | This message is produced when the firmware is tested. It should never be seen during normal operation. Diagnostic Report^ |
| MCREST#14 Err. Rcvd Line. Does not begin STX 0x%02x %c | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#15 Err. Rcvd Line bad format | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#16 FYI System Trouble Strings loaded from file=systrb.ini | The driver allows the list of System Trouble strings to be updated. The procedure is outline in this manual. When they are loaded from the file "systrb.ini" then this warning is printed. |
| MCREST#09 FYI AddSysTrb=%03d <%s> | When System Trouble strings are loaded from the file "systrb.ini" this message is printed each time a new string is added. |
| MCREST#17 Err No space to add SysTrb  Max=%d (%d) | There is only space for 250 System Trouble strings to be loaded. Check the file "systrb.ini"  and remove some as required. |
| MCREST#18 FYI Classifications loaded from file=classi.ini | The driver allows the list of Classifications to be updated. The procedure is outline in this manual. When they are loaded from the file "classi.ini" then this warning is printed. |
| MCREST#18 FYI AddClassi=%03d <%s> | When Classifications are loaded from the file "classi.ini" this message is printed each time a new string is added. |
| MCREST#20 Err No space to add classi.ini Max=%d (%d) | There is only space for 250 Classifications strings to be loaded. Check the file "classi.ini"  and remove some as required. |

| | |
|---|---|
| MCREST#21 Err. Classification=<%s> not known | An event reported from the FACP has a Classification that is not recognized by the list of Classifications in use. The list can be modified. |
| MCREST#22 Err. Cant find DA=%s to store MultiStateTrbls | The Data Array name specified in the Map Descriptor field "Multicrest_MultiTrbls_DA" has been specified but the driver cannot find that Data Array. This message is printed a maximum of 10 times and then suppressed. Correct CSV* |
| MCREST#23 FYI DA=%s Off=%d MultiVal=%d MultiTrb%s | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#24 Err. Cant find DA=%s to store MultiStateAlarm | The Data Array name specified in the Map Descriptor field "Multicrest_MultiActive_DA" has been specified but the driver cannot find that Data Array. This message is printed a maximum of 10 times and then suppressed. Correct CSV* |
| MCREST#25 FYI DA=%s Off=%d MultiVal=%d MultiAlm=%s | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#26 Err. Cant find any MD's to store data. MCREST#26 Err. Node=%d Loop=%d Dev=%d %s | An event whose address is reported has been received but with the there is no place to store an event with this address. Correct CSV* |
| MCREST#16 Err. Node=%d Loop=%d Dev=%d %s | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#27 Err. Mismatch Nodes %d %d | This message is produced when the firmware is tested. It should never be seen during normal operation. Diagnostic Report^ |
| MCREST#28 FYI %s DA=%s Off=%d St=%d | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#29 Err. Cant find DA=%s to store Summary | The Data Array name specified in the Map Descriptor field "Multicrest_Summary_DA" has been specified but the driver cannot find that Data Array. This message is printed a maximum of 10 times and then suppressed. Correct CSV* |
| MCREST#30 FYI Descriptor=%s | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#30 FYI DA=%s Off=%d val=0x%04x Bit=%d Before | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#31 FYI DA=%s Off=%d val=0x%04x After | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#33 FYI Descriptor=%s | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#33 FYI DA=%s Off=%d val=0x%04x Bit=%d Before | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#34 FYI DA=%s Off=%d val=0x%04x After | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#35 Err. Cant find any place to store - %s MCREST#35 Err. Node=%d Dev=%d Ckt=%d Loop=%d. | An event whose address is reported has been received but with the there is no place to store an event with this address. Correct CSV* |
| MCREST#36 Err. Node=%d Cant find DA to store SysTroubles | A System or Panel Trouble has been reported. There is no map descriptor for the node specified where the Multicrest_Function is either "System_Trouble" or "Panel_Troubles". Correct CSV* |
| MCREST#37 FYI DA=%s Node=%d Off=%d St=%d Trb=%s | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |

| | |
|---|---|
| MCREST#38 Err. Cant recognize System Trouble String<br>MCREST#38 Err. string=<%s> | The Panel reported a system Trouble event but the Event string was not recognized. A list of default troubles and a procedure to add new ones is provided in this manual. Correct CSV* |
| MCREST#39 Err. Cant find any MD's to store data. | There don't appear to be any map descriptors specified for the Multicrest driver. Read the section on Map Descriptors. Correct CSV* |
| MCREST#27 Err. Mismatch Nodes %d %d | This message is produced when the firmware is tested. It should never be seen during normal operation. Diagnostic Report^ |
| MCREST#40 Err. Node=%d Sys/PanelTrouble No Storage. | A System or Panel Trouble has been reported. There is no map descriptor for the node specified where the Multicrest_Function is either "System_Trouble" or "Panel_Troubles" Correct CSV* |
| MCREST#41 FYI Reset time start=%ld | A system reset has been received from the panel. The driver notes how long the reset takes to complete (clearing relevant data arrays). |
| MCREST#42 Err. Cant find any MD's to store data. | There don't appear to be any map descriptors specified for the Multicrest driver. Read the section on Map Descriptors. Correct CSV* |
| MCREST#43 FYI. Classifications retained after reset. | Informs you that a system reset does not clear the classification data (if it is being stored.) This message is printed once only. |
| MCREST#45 Err. Node=%d System Reset - No DA's. | A system reset has been received from the panel. There were no data arrays to clear. |
| MCREST#46 FYI Reset time end=%ld delta=%ld | A system reset has been received from the panel. The driver notes how long the reset takes to complete (clearing relevant data arrays). |
| MCREST#47 Err. No place to store SysTroubles for panel=%d | A System or Panel Trouble has been reported. There is no map descriptor for the node specified where the Multicrest_Function is either "System_Trouble" or "Panel_Troubles" Correct CSV* |
| MCREST#48 FYI DA=%s Node=%d Off=%d St=%d Trb=%s | This message is produced in normal operation when a diagnostic feature has been enabled. Read the manual to find out how the diagnostic can be disabled. |
| MCREST#49 Err. Cant recognize System Trouble String<br>MCREST#49 Err. string=<%s> | The Panel reported a system Trouble event but the Event string was not recognized. A list of default troubles and a procedure to add new ones is provided in this manual. Correct CSV* |
| MCREST#51 FYI. Use an Array called <%s> to expose diagnostic info. | Read the section named "Exposing Driver Stats" |

## Appendix A.8    Exposing Driver Stats

The driver makes some of its operating statistics available in a Data Array where they can be read by a remote client. The lines from the example below can be cut and pasted into a configuration file.

| Data_Arrays, | | |
| --- | --- | --- |
| Data_Array_Name, | Data_Format, | Data_Array_length, |
| MulticrestStats, | UINT32, | 1000, |

| OFFSET | DESCRIPTION |
| --- | --- |
| 0 | Increments each time the driver receives a complete message from the panel |
| 1 | Increments each time a message is received that the driver does not have the capability of processing. |
| 2 | Increments each time a system trouble events is processed and the driver does not recognize the system trouble string (event). It may require that you update the system trouble file. |
| 3 | Increments each time a 'System Initialized' message is received. |
| 4 | Increments each time a 'System Normal' message is received. |
| 5 | Increments each time a 'System Trouble' message is received. |
| 6 | Increments each time a 'Representative Trouble' message is received. |
| 7 | Increments each time a 'Drill' message is received. |
| 8 | Increments each time a 'Silence' message is received. |
| 9 | Increments each time a 'System Reset' message is received. |
| 10 | Increments each time a 'Module Disable' message is received. |
| 11 | Increments each time a 'NAC Disable' message is received. |
| 12 | Increments each time a 'CMS Disable' message is received. |
| 13 | Increments each time a 'Output Disable' message is received. |
| 14 | Increments each time a 'Alert' message is received. |
| 15 | Increments each time a 'Action' message is received. |
| 16 | Increments each time a 'Alarm' message is received. |
| 17 | Increments each time a 'Alarm F1' message is received. |

| OFFSET | DESCRIPTION |
|---|---|
| 18 | Increments each time a 'Alarm F2' message is received. |
| 19 | Increments each time a 'Confirm' message is received. |
| 20 | Increments each time a 'Verify' message is received. |
| 21 | Increments each time a 'Supervisory' message is received. |
| 22 | Increments each time a 'No Answer' message is received. |
| 23 | Increments each time a 'Comm Fault' message is received. |
| 24 | Increments each time a 'Disable' message is received. |
| 25 | Increments each time a 'Type Error' message is received. |
| 26 | Increments each time a Fault' message is received. |
| 27 | Increments each time a 'Data Error' message is received. |
| 30 | Increments System Trouble String was not recognized. |
| 31 | Increments each time a System Trouble report cannot be stored – typically a consequence of the message being ignored because system troubles for the node are not configured. |
| 32 | Increments each time a device event message could not be stored because no storage was defined for the Node, Loop or the event type |
| 33 | Increments If the driver attempts to store summary data but the Data array could not be found. |
| 34 | Increments when there is no MD to store an event. |
| 35 | Increments when there are no MD's to store a system trouble. |
| 36 | Increments each time the classification was not recognized. |
| 37 | Debug Mode. Set the value of this location to 1 to have the driver dump additional debug info into the log. Set value to 3 for even more and 7 for max. |
| 38 | Like #33 but for a full summary. |
| 39 | Increments each time a message is received which reports a point as Active |
| 40 | Increments each time a message is received which reports a point as Inactive |
| 41 | Increments each time a message is received which reports a Device Bypass |
| 42 | Increments each time a message is received which reports a  Missing Device |
| 43 | Increments each time a message is received which reports a  Missing Device |
| 44 | Increments each time a message is received which reports a  "Multiple Address" (This is a Multicrest term) |

| OFFSET | DESCRIPTION |
|--------|-------------|
| 45 | Increments each time a message is received which reports a "Low Temp" (This is a Multicrest term) |
| 46 | Increments each time a message is received which reports a "Sensor Fault" (This is a Multicrest term) |
| 47 | Increments each time a message is received which reports a "Discharge Cnf" (This is a Multicrest term) |
| 48 | Increments each time a message is received which reports a "Pre Discharge" (This is a Multicrest term) |
| 49 | Increments each time a message is received which reports a "Discharge" (This is a Multicrest term) |
| 50 | This is used to control the behavior of the driver. If the (default) value = 0 then a System Reset message from any node will reset the data for all nodes. If the value = 1 then the driver only resets the data for the node reporting the System Reset. |