



**FieldServer**  
**FS-8700-39 EST3-ECP**  
**Driver Manual**  
(Supplement to the FieldServer Instruction Manual)

**APPLICABILITY & EFFECTIVITY**

Effective for all systems manufactured after August 2016.

Driver Revision: 1.13  
Document Revision: 10.A

## Technical Support

Please call us for any technical support needs related to the FieldServer product.

Sierra Monitor Corporation  
1991 Tarob Court  
Milpitas, CA 95035

Website: [www.sierramonitor.com](http://www.sierramonitor.com)

U.S. Support Information:

+1 408 262-6611

+1 800 727-4377

Email: [support@sierramonitor.com](mailto:support@sierramonitor.com)

EMEA Support Information:

+44 2033 1813 41

Email: [support.emea@sierramonitor.com](mailto:support.emea@sierramonitor.com)

**TABLE OF CONTENTS**

**Table of Contents** ..... 3

**1 EST3 Description** ..... 5

**2 Driver Scope of Supply** ..... 5

    2.1 Supplied By Sierra Monitor for this Driver ..... 5

    2.2 Provided By the Supplier of 3<sup>rd</sup> Party Equipment ..... 5

        2.2.1 *Hardware* ..... 5

        2.2.2 *Required 3<sup>rd</sup> Party Configuration* ..... 5

**3 Hardware Connections**..... 6

    3.1 Pinouts ..... 6

    3.2 Connection Notes ..... 7

**4 Data Array Parameters** ..... 8

**5 Configuring the FieldServer as a EST3 Client** ..... 9

    5.1 Client Side Connection Parameters ..... 9

    5.2 Client Side Node Parameters ..... 10

    5.3 Client Side Map Descriptor Parameters ..... 10

        5.3.1 *FieldServer Related Map Descriptor Parameters* ..... 10

        5.3.2 *Driver Related Map Descriptor Parameters* ..... 11

        5.3.3 *Map Descriptor Example 1* ..... 13

        5.3.4 *Map Descriptor Example 2* ..... 13

        5.3.5 *Map Descriptor Example 3* ..... 13

**Appendix A. Useful Features** ..... 14

    Appendix A.1. Panel Masks ..... 14

    Appendix A.2. Setting up Multiple Parameters ..... 14

    Appendix A.3. Priority ..... 14

    Appendix A.4. Report and Delta Service ..... 14

        Appendix A.4.1. Report Service (regular polling) ..... 14

        Appendix A.4.2. Delta Service (unsolicited messages) ..... 14

            Appendix A.4.3. *Example Configuration using Delta Mode only* ..... 15

    Appendix A.5. Optional Data Arrays ..... 16

        Appendix A.5.1. *Relay Confirmation State* ..... 16

        Appendix A.5.2. *EST\_Types\_DA* ..... 16

        Appendix A.5.3. *EST\_Types\_Update* ..... 16

        Appendix A.5.4. *DA\_Bit\_Name* ..... 17

        Appendix A.5.5. *EST\_Report\_Type* ..... 17

    Appendix A.6. Object Types and Required Parameters ..... 18

    Appendix A.7. Set Date and Time To Panel ..... 19

**Appendix B. Troubleshooting** ..... 20

    Appendix B.1. Multiple Com Errors ..... 20

**Appendix C. Reference** ..... 21

    Appendix C.1. EST3 Data Format ..... 21

        Appendix C.1.1. *Input data format* ..... 21

        Appendix C.1.2. *Output data format* ..... 24

    Appendix C.2. EST3 ECP Connection Statistics ..... 25

    Appendix C.3. Driver Error Messages ..... 26

**LIST OF FIGURES**

Figure 1: FieldServer to ETS3 Hardware Connections..... 6  
Figure 2: QuickServer to ETS3 Hardware Connections ..... 7

## 1 EST3 DESCRIPTION

The EST3 External Communications Protocol (ECP) driver allows the FieldServer to transfer data to and from Edwards EST devices over RS-232 or RS-485 (with converter) serial ports using the EST3 ECP protocol. In the EST application the FieldServer always emulates a Client.

The FieldServer makes use of the following ECP Services.

**Read Operations:**

- Report Service
- Delta Service

**Write Operations:**

- Command Service

All read points are continually updated using the Report Service; response times are enhanced using Delta Service broadcasts by the EST3 panel.

## 2 DRIVER SCOPE OF SUPPLY

### 2.1 Supplied By Sierra Monitor for this Driver

PART #	Description
FS-8915-10	7-ft Cat5 cable with RJ45 connectors at both ends
FS-8917-16	Cable, EIA232:485 Pigtail RJ45 Port

### 2.2 Provided By the Supplier of 3<sup>rd</sup> Party Equipment

#### 2.2.1 Hardware

Check to make sure that RS232 daughter board is installed on the CPU board.

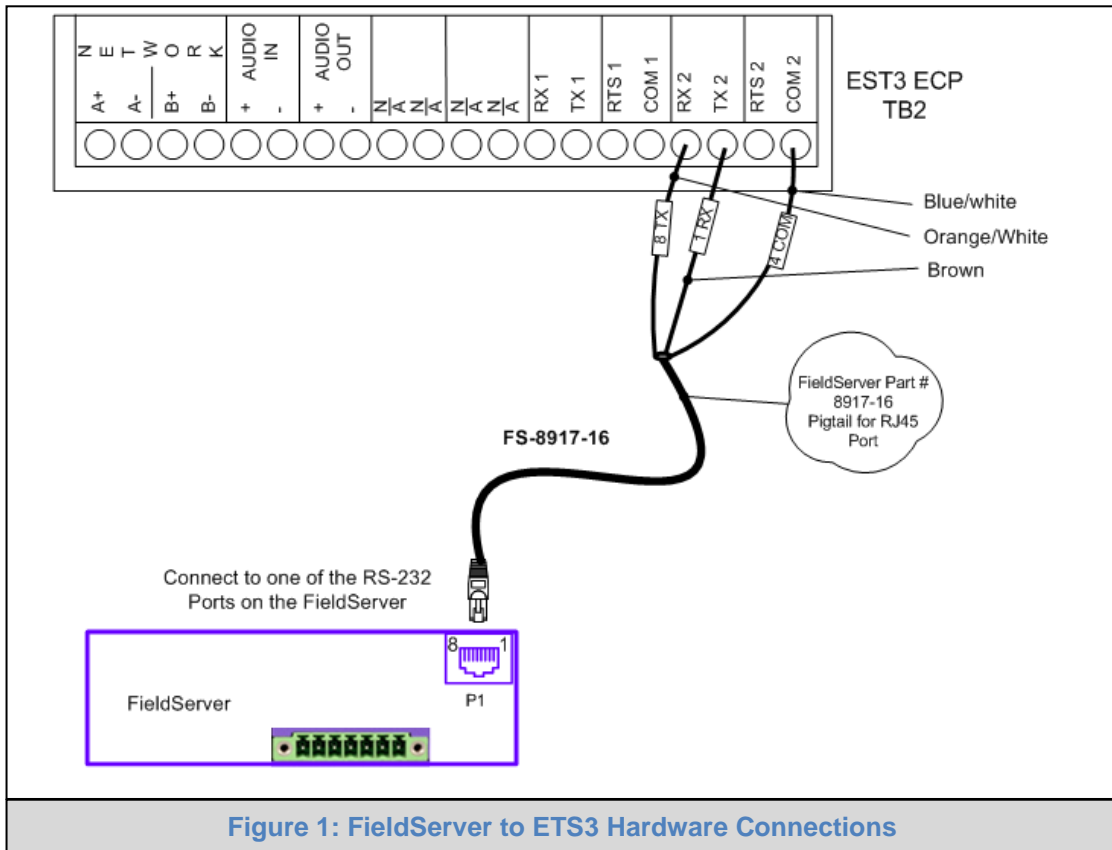
PART #	Description
	EST3 Gateway
3-RS232 Card Rev 5.0	Needs to be installed for PAL chip Rev 1.1.
3-RS232 Card Rev 6.0	Needs to be installed for PAL chip Rev 1.2.

#### 2.2.2 Required 3<sup>rd</sup> Party Configuration

The EST3 panel has to be set into the “Gateway Mode” using the EST3 Configuration tool (3-SDU).

Possible Protocols	
ECP Type II	Supported
ECP Type II with Text	Not Supported
ECP Type III	Supported
ECP Type III with Text	Not Supported

### 3 HARDWARE CONNECTIONS



#### 3.1 Pinouts

RJ-45 Pin	From RJ-45	Color	To EST3 ECP TB2
1	RX	Brown	TX 2
8	TX	White/Orange	RX 2
4	COM	Blue/white	COM 2

### 3.2 Connecting to QuickServer

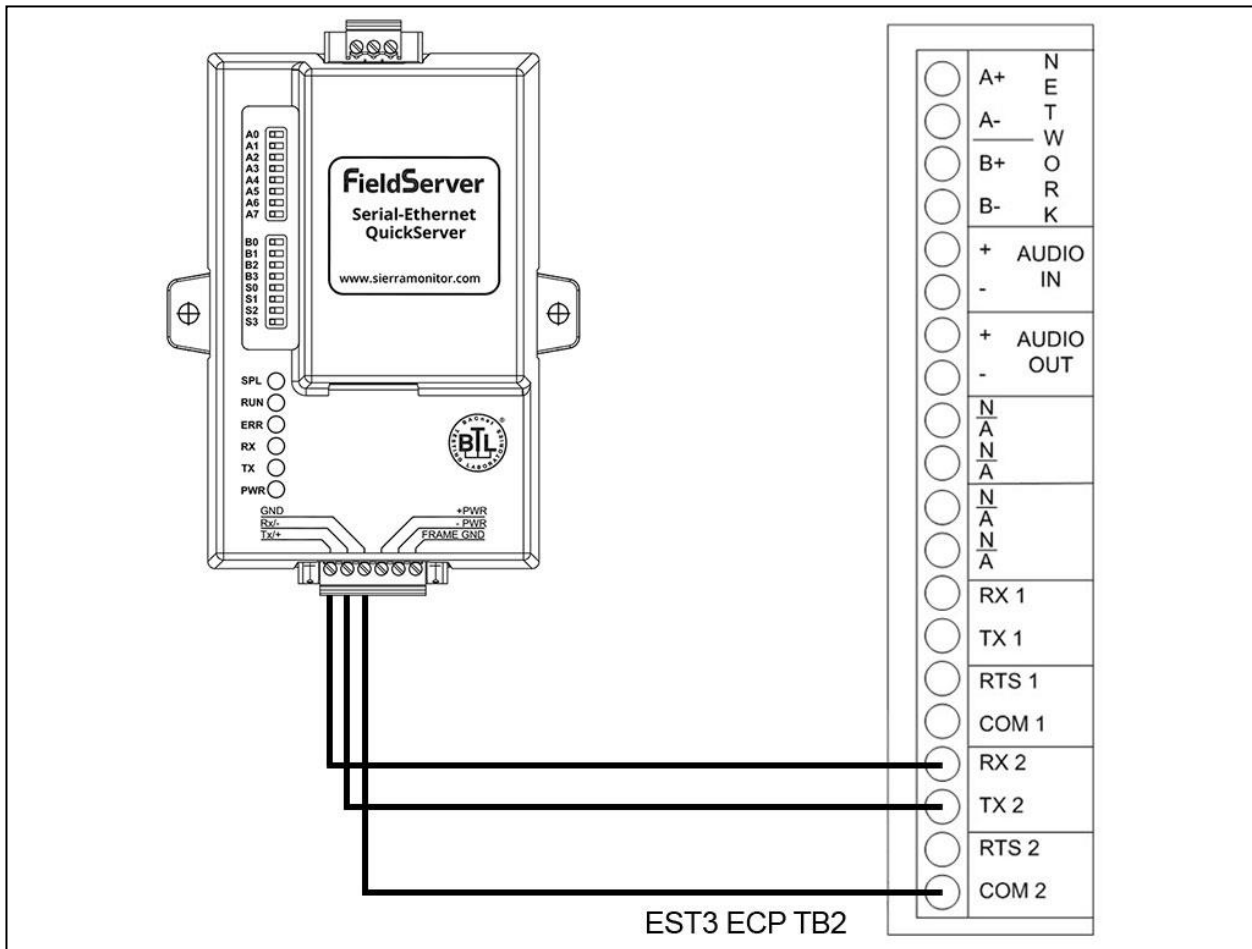


Figure 2: QuickServer to ETS3 Hardware Connections

### 3.3 Connection Notes

- There is normally only one interface to an EST3 system, even if multiple panels exist. All data for the entire system is received through this single interface.
- The interface on the EST3 panel may be COM1<sup>1</sup> or COM2, but it needs to be configured in the EST3 panel as a gateway. Gateway Type III is recommended.
- The EST3 panel monitors the quality of communications to the FieldServer and will detect a poor connection. Poor connections are normally caused by inappropriate selection of gateway type, but can also be related to installation quality.

<sup>1</sup> For single port operation, Port 1 may not operate correctly on system startup. This may be reported as a communications error, or may simply result in communication problems with the RS-232 port. 3-CPU3 units with date codes 07120 or later do not exhibit this issue because of the version of firmware used. No action is required. 3-CPU3 with date codes earlier than 07120 may exhibit this issue. Units that do exhibit the issue must be replaced with newer units.  
Date code format is: YYDDD For example, 12145, that is 2012 May 24 (145th day of the year).

## 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.	Uint16
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_Array_Format , Data_Array_Length
DA_AI_01        , Float              , 200
DA_AO_01        , Float              , 200
DA_DI_01        , Bit                , 200
DA_DO_01        , Bit                , 200
    
```



## 5 CONFIGURING THE FIELDSEVER AS A EST3 CLIENT

For detailed information on FieldServer configuration, 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 with an EST3 Server.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for EST3 communications, 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: In the tables below, \* indicates an optional parameter, with the bold legal value being the default.**

### 5.1 Client Side Connection Parameters

Section Title		
Connections		
Column Title	Function	Legal Values
Port	Specify which port the device is connected to the FieldServer.	P1-P2, R1-R2 <sup>2</sup>
Baud	Specify baud rate.	9600, <b>19200</b> <sup>3</sup>
Parity*	Specify parity.	<b>None</b>
Data_Bits*	Specify data bits.	<b>8</b>
Protocol	Specify protocol to be used by this port.	EST3

**Example**

```

// Client Side Connections

Connections
Port          , Baud    , Parity  , Data_Bits , Protocol
P1            , 19200  , None   , 8         , EST3
    
```

<sup>2</sup> Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

<sup>3</sup> Edwards Recommends 19200-baud minimum in order to prevent buffer overruns in the EST3 panel.

## 5.2 Client Side Node Parameters

Section Title		
Nodes		
Column Title	Function	Legal Values
Node_Name	Provide name for Node.	Up to 32 alphanumeric characters
Node_ID	MAC ID of Panel.	1 – 24 (normally 1)
Protocol	Specify protocol used.	EST3
Port	Specify which port the device is connected to the FieldServer.	P1-P2, R1-R2 <sup>2</sup>
Timeout	Specify time allowed between poll and response.	≥10 seconds
IC_Timeout	The IC_Timeout parameter monitors the time between characters in a response. If the time exceeds the IC_Timeout, the response is discarded and is considered a Timeout.	0-0.5s, <b>0.05s</b> , recommended 0.5s
EST_MX_Read_Points*	Specify the maximum number of points that can be polled in single request. Legacy panel firmware (prior to spec version 5.20) supports a maximum of 34, but new firmware supports a maximum of 15.	<b>1-34</b>
EST_StatusTime_Format*	Specify the format to store status time in EST_StatusTime_DA Data Array.	yyyy, mm, dd, hh, mm, <b>ss</b>
Est_Independent_Events*	Specify if events from panel should be considered independent of each other or not. If events are independent, the driver will store and clear any event on receiving a message from the panel. If events are not independent, the driver will store the current event and clears all other events for the point.	Yes, <b>No</b>

### Example

```
// Client Side Nodes

Nodes
Node_Name , Node_ID , Protocol , Port , Timeout , IC_Timeout , EST_Mx_Read_Points
EST_1 , 1 , EST3 , P1 , 10s , 0.5s , 15
```

## 5.3 Client Side Map Descriptor Parameters

### 5.3.1 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 <b>Section 4</b> .
Data_Array_Offset	Starting location in Data Array.	0 to maximum specified in <b>Section 4</b> .
Function	Function of Client Map Descriptor.	RDBC, WRBC <sup>4</sup> , WRBX

<sup>4</sup> WRBC is not recommended, as startup values may be invalid.

**5.3.2 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 <b>Section 5.2</b> .
EST_Panel	Panel	0 - 63
EST_Card	Card	0 - 255
Address	Starting address of read block.	0 - 2127
Length	Length of Map Descriptor. Longer map descriptors (where possible) are better from a performance point of view, as well as from a memory use point of view, since each map descriptor comes with a memory overhead as well as a protocol handshaking overhead.	Reads: 1 to maximum address of panel and card. A length of up to 34 is recommended. Older panels do not accept odd numbers. Writes: 1
EST_Object_Type	Object type parameter.	Relay, Led, ServGroup, GuardPtrl, ChkInGrp, AltMessage, Reset, PanelSil, AlarmSil, Drill, GAIInhibit, Reboot, Evacuate, AltSensitiv, Partition, CmdList, Input, Zone, Card, AndGroup, Matrix, TimeCtrl, InstructionText, AmpPower, AudMsg, CmdList, MCmsg, SetDate, SetTime
EST_Index	Index parameter	0 - 65535
EST_Route	Route parameter	0 - 255
EST_Priority	Pri parameter	0 - 4 (Refer to Appendix A.3)
EST_Chan	Chan parameter	0 - 255
EST_Msg_Index	MsgIndex parameter	0 - 65535
EST_Account	Account parameter	0 - 255
EST_ConfCL	ConfCL parameter	0 - 65535
EST_Panel_Mask_1	PanelMask(1) parameter (LSB)	0 - 255
EST_Panel_Mask_2	PanelMask(2) parameter	0 - 255
EST_Panel_Mask_3	PanelMask(3) parameter	0 - 255
EST_Panel_Mask_4	PanelMask(4) parameter	0 - 255
EST_Panel_Mask_5	PanelMask(5) parameter	0 - 255
EST_Panel_Mask_6	PanelMask(6) parameter	0 - 255
EST_Panel_Mask_7	PanelMask(7) parameter	0 - 255
EST_Panel_Mask_8	PanelMask(8) parameter (MSB)	0 - 255
EST_Types_Update*	Update the default Types list or refer to new list. Requires 'EST_Types_DA' parameter to be defined (refer to <a href="#">Appendix A.5.3</a> ).	-, <b>yes</b> , new list
Data_Array_Low_Scale*	Scaling zero in Data Array	-32767 to 32767, <b>0</b>
Data_Array_High_Scale*	Scaling max in Data Array	-32767 to 32767, <b>100</b>
Node_Low_Scale*	Scaling zero in Connected Node	-32767 to 32767, <b>0</b>

Column Title	Function	Legal Values
Node_High_Scale*	Scaling max in Connected Node.	-32767 to 32767, <b>100</b>
EST_Status_DA	Data Array to store Point Status (refer to example in <b>Section 5.3.5</b> ).	One of the Data Array Names declared in <b>Section 4</b> .
EST_Alarm_DA*	Data Array to store Alarm state. <sup>5</sup>	
EST_Trouble_DA*	Data Array to store Trouble state. <sup>5</sup>	
EST_Supervisory_DA*	Data Array to store Supervisory state. <sup>5</sup>	
EST_Monitor_DA*	Data Array to store Monitor state. <sup>5</sup>	
EST_COActive_DA*	Data Array to store CO active state. <sup>5</sup>	
EST_Others_DA*	Data Array to store state other than Alarm, Trouble, Supervisory or Monitor. <sup>5</sup>	
EST_Active_DA*	Data Array to store the raw value of the active bit.	
EST_Relay_Conf_DA*	Relay confirmation state. Refer to Appendix A.5 for more information.	
EST_Types_DA*	Data Array to hold Types category (refer to <b>Appendix A.6</b> ).	
DA_Bit_Name*	Acknowledge State (refer to <b>Appendix A.5.4</b> ).	
EST_Report_Type*	Store information from particular Report	-, <b>Point I/O Status</b> , Sensitivity, Date Time
EST_Store_Type*	<p>This parameter is applicable only to Map Descriptors where EST_Report_Type is 'Point I/O Status'.</p> <p>If this parameter is not defined and/or the value is other than 'Status', the Driver will store raw 16 Bit data in the Data Array pointed to by the Data_Array_Name parameter of the Map Descriptor.</p> <p>If the value of this parameter is 'Status' then the Driver will store the most recent status of the point. Since the Data_Array_Name Data Array and the Data Array under EST_Status_DA will hold the same data, EST_Status_DA may be omitted (refer also to <b>Section 5.3.5</b>).</p>	<p><b>Status</b></p> <p>All other values will be ignored and will not have any affect.</p>
EST_StatusTime_DA*	<p>Data Array to store FieldServer's time when Point Status changes.</p> <p>By default or when Node parameter EST_StatusTime_Format is set to ss, driver will store FieldServer's Time in seconds since 1 Jan 1970.</p> <p>If EST_StatusTime_Format is yyyyymmddhhmmss driver will store time as year, month (1-12), date (1-31) hour (0-23) minutes (0-59) seconds (0-59) in 6 sequential data array locations.<sup>6</sup></p>	One of the Data Array Names declared in <b>Section 4</b> .

<sup>5</sup> Refer to **Appendix A.4** and **Section 5.3.4**.

<sup>6</sup>Timestamp will be recorded only when MD is configured to store Point Status i.e. either EST\_Status\_DA is used or EST\_Store\_Type is set to 'Status'.

### 5.3.3 Map Descriptor Example 1

```
// Client Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name      , Data_Array_Name      , Data_Array_Offset      , Function      , Node_Name      , EST_Panel
MD_LED_1                 , DA_AI_01                 , 0                       , Rdbc          , EST_1          , 1

, EST_Card      , Address      , Length      , EST_Store_Type      , Scan_Interval
, 1             , 0            , 20          , Status              , 5.0s
```

### 5.3.4 Map Descriptor Example 2

This Map Descriptor will cause the specified EST point to be read every 50 seconds (subject to throughput constraints); the full 16-bit status values received from regular polls will be stored in Data Array DA\_AI\_01 at offsets 0 – 19, and the Alarm, Trouble, Supervisory, Monitor or Other status (received by regular poll, or Delta message) will be stored in Data Arrays TROUBLES and ALARMS, SUPERVISORY, MONITOR, OTHERS also at offsets 0 – 19. Refer also to [Appendix C](#).

```
// Client Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name      , Data_Array_Name      , Data_Array_Offset      , Function      , EST_Trouble_DA      , EST_Alarm_DA
MD_LED_1                 , DA_AI_01                 , 0                       , Rdbc          , TROUBLES           , ALARMS

, Node_Name      , EST_Panel      , EST_Card      , Address      , Length      , Scan_Interval
, EST_1          , 1              , 1             , 0            , 20          , 50s
```

### 5.3.5 Map Descriptor Example 3

This Map Descriptor will cause the specified EST point to be read every 50 seconds (subject to throughput constraints). The full 16-bit status values received from regular polls will be stored in Data Array DA\_AI\_01 at offsets 0 – 19, and the *Classification* as Alarm, Trouble, Supervisory, Monitor , Other, Relay Confirmation, COActive, Pre-Alarm, Disabled or Test status in the Data Array STATUS at offsets 0 – 19. Refer also to [Appendix C.1](#).

The classification values are as follows:

Normal	0	Relay confirmation	6
Alarm	1	CO Active	7
Supervisory	2	Pre-Alarm	8
Trouble	3	Disabled	9
Monitor	4	Test	10
Other	5		

```
Map_Descriptors
Map_Descriptor_Name      , Data_Array_Name      , Data_Array_Offset      , Function      , EST_Status_DA      , Node_Name
MD_LED_1                 , DA_AI_01                 , 0                       , Rdbc          , STATUS              , EST_1

, EST_Panel      , EST_Card      , Address      , Length      , Scan_Interval
, 1             , 1            , 0            , 20          , 50s
```

## Appendix A. USEFUL FEATURES

### Appendix A.1. Panel Masks

The 8 PanelMask parameters combine to form a 64-bit field in which each bit denotes an EST panel in the network. Commands that use the Panel Masks are executed by each panel for which the corresponding bit is set. The Map Descriptor need only define non-zero bytes in the PanelMask. Typically this will be EST\_Panel\_Mask\_1 with a value of 1.

### Appendix A.2. Setting up Multiple Parameters

The multiple parameters required by a command must be stored in consecutive locations in the Data Array specified by the Map Descriptor, starting at the specified Data\_Array\_Offset. All parameters must be set up before the EST command is executed. This is done using a Block Write (if supported by the protocol originating the data). All parameters are set up in a single write operation, and the EST command is triggered with all values in place.

### Appendix A.3. Priority

The write priority has the possible values of Set, Latch, Low, Medium, and High. These are encoded as 00h, 01h, 02h, 03h and 04h respectively.

### Appendix A.4. Report and Delta Service

The EST ECP3 Report and Delta services report point status slightly differently. The FieldServer EST3 driver deals with these differences as follows:

#### Appendix A.4.1. Report Service (regular polling)

The full 16-bit point status is stored to the Read Map Descriptor's Data Array. If optional Data Arrays are configured, then the point status is also interpreted according to its Alarm, Trouble, Supervisory or Monitor status and the corresponding Data Array is updated. See [Appendix A.5](#) for details.

#### Appendix A.4.2. Delta Service (unsolicited messages)

Delta Service messages do not provide the same 16-bit status value as supplied by the Report Service. For this reason, ONLY the optional Data Arrays will be updated when a Delta message is received. Delta messages will therefore only have an effect when optional Data Arrays have been configured. The delta service will be sent out through ECP automatically. There's no panel setting for this. The EST3 panel always sends a delta frame when there is a state change. Refer also to [Appendix A.5](#) for additional details.

Appendix A.4.3. Example Configuration using Delta Mode only

**NOTE: It is recommended that at least one active Map Descriptor is defined to alert the Driver if the connection fails. As many passive Clients as desired can then be appended.**

Ports	
Port	, Baud
P1	, 9600

Nodes				
Node_Name	, Node_ID	, Port	, Protocol	, Timeout
Node_A	, 1	, P1	, EST3	, 4s

Data_Arrays		
Data_Array_Name	, Data_Format	, Data_Array_Length
DA_TEST1	, Uint16	, 100
DA_TROUBLE	, Bit	, 40
DA_ALARM	, Bit	, 40

Map_Descriptors					
Map_Descriptor_Name	, Scan_Interval	, Data_Array_Name	, Data_Array_Offset	, Function	, Node_Name
MD_TEST1	, 10.0s	, DA_TEST1	, 0	, Rdbc	, Node_A
MD_TEST2	, -	, DA_TEST1	, 20	, Passive_Client	, Node_A

, EST_Panel	, EST_Card	, Address	, Length	, EST_Trouble_DA	, EST_Alarm_DA
, 1	, 1	, 0	, 20	, DA_TROUBLE	, DA_ALARM
, 7	, 3	, 130	, 20	, DA_TROUBLE	, DA_ALARM

### Appendix A.5. Optional Data Arrays

When optional parameters (EST\_Alarm\_DA, EST\_Trouble\_DA, EST\_Supervisory\_DA, EST\_Monitor\_DA, or EST\_COActive\_DA) are used, the driver automatically separates the Alarm Trouble, Supervisory, Monitor, CO Active bit from the incoming EST message and places the bit(s) in these Data Arrays at the same offset as the incoming message. Other types can be stored in EST\_Others\_DA Data Array. Only one Data type will be reported as active at a particular time; i.e. if Alarm is reported Driver will set Alarm bit and reset corresponding bits from Trouble, Supervisory and Monitor Data Arrays. The Driver does not correlate other types with Alarm, Trouble, Supervisory or Monitor types. For different point types the active bit has a different meaning. If the user is interested in the meaning, the EST\_Supervisory\_DA Data Array should be configured. If the user is interested in the raw value of the Active bit, the data array EST\_Active\_DA should be configured.

#### Appendix A.5.1. Relay Confirmation State

The driver can store the relay confirmation state of a point in a Data Array under the parameter EST\_Relay\_Conf\_DA parameter.

0 – Relay state is not confirmed;                                            1 – Relay state confirmed.

#### Appendix A.5.2. EST\_Types\_DA

If it is necessary to override the default Types and Category list ([Appendix C.1.1](#)) the EST\_Types\_DA parameter should be used.

For example, to put HEAT type into the “Monitor” category where all other defaults remain the same, preload the data array as follows:

Preloads			
Data_Array_Name	,Preload_Data_Index	,Preload_Data_Value	,Length
DA_Types	,2	,4	,1

Preload\_Data\_Index is the type of device and Preload\_Data\_Value holds the category for the specified device type. The legal values for Preload\_Data\_Value are listed below:

Alarm	1
Supervisory	2
Trouble	3
Monitor	4
Others	5
CO Active	7

#### Appendix A.5.3. EST\_Types\_Update

If this parameter is not defined or is “yes”, the driver will update its internal driver list according to the parameters preloaded under EST\_Types\_DA. If the parameter is set to “New List”, the driver will use ONLY the parameters defined under EST\_Types\_DA and will not reference or update the default list at all. The driver will therefore not store any data for types that do not have preloaded categories.



#### Appendix A.5.4. DA\_Bit\_Name

The “Acknowledge<sup>7</sup>” state of a point is stored under this Data Array if configured.

- 0 – Acknowledged or Acknowledge not required.
- 1 – Point is not Acknowledged yet. To Acknowledge the point change the value to “0”.

#### Appendix A.5.5. EST\_Report\_Type

The dirtiness of a device can be stored as a value \*10 for System Sensor Devices and as the actual value for other Device Types (the panel sends % for other Device Types).

Map_Descriptors					
Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Function	,Node_Name	,EST_Report_Type
MD_Poll_Sensitivity	,DA_Dirtiness	,0	,Rdbc	,EST_1	,Sensitivity

,EST_Panel	,EST_Card	,Address	,Length	,Scan_interval
,1	,2	,1	,3	,5.0s

<sup>7</sup> The EST3 panel is only able to send/accept Acknowledgement when the panel is in Proprietary Mode. Contact EST (Edwards Systems Technology) for information on configuring the panel.

**Appendix A.6. Object Types and Required Parameters**

The tables below list the fields required for the Map Descriptor implementing each object type.

<b>Object Type</b>	<b>Required Parameters</b>
<b>Activate and Restore Commands (01h and 11h)</b>	
Relay	EST_Panel, EST_Card, Address, EST_Priority
Led	EST_Panel, EST_Card, Address, EST_Priority
ServGroup	EST_Index
GuardPtrl	EST_Index, EST_Route
ChkInGrp	EST_Index
AltMessage	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
AmpPower	EST_Panel, EST_Card, EST_Priority, EST_Chan
AudMsg	EST_Panel, EST_Card, EST_Msg_Index, EST_Priority, EST_Chan
Reset	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
PanelSil	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
AlarmSil	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
Drill	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
GAINhibit	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
Reboot	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
Evacuate	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
AltSensitiv	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
MCmsg	EST_Panel, EST_Card, EST_Priority, EST_Account, EST_ConfCL
CmdList	EST_Index
AndGroup	EST_Index
SetDate	EST_Object_Type; refer to Appendix A.7
SetTime	EST_Object_Type; refer to Appendix A.7
ClrPartitionEvent	EST_Index
DeviceTest	EST_Panel, EST_Card, Address
SysFunction1	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
SysFunction2	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
SysFunction3	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
SysFunction4	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
CalibrateAnalog	EST_Panel, EST_Card, Address
2StageSounder	EST_Panel, EST_Card, Address, EST_Priority
SensorByPass	EST_Panel, EST_Card, Address
GasAcceleratedResponse	EST_Panel, EST_Card, Address
RemoteReadLock	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
RemoteWriteUnlock	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
LogicalOutput	Address, EST_Priority
LampTest2	EST_Panel_Mask(1..8); refer <a href="#">Section 5.3.2</a>
<b>Enable and Disable Commands (02h and 12h)</b>	
Relay	EST_Panel, EST_Card, Address
Led	EST_Panel, EST_Card, Address
Input	EST_Panel, EST_Card, Address
Zone	EST_Index
Card	EST_Panel, EST_Card
ServGroup	EST_Index
AndGroup	EST_Index

Object Type	Required Parameters
Matrix	EST_Index
TimeCtrl	EST_Index
GuardPtrl	EST_Index, EST_Route
ChkInGrp	EST_Index
AmpPower	EST_Panel, EST_Card
InstructionText	EST_Panel, EST_Card, Address
MCmsg	EST_Panel, EST_Card, Address
CmdList	EST_Index
2StageSounder	EST_Panel, EST_Card, Address
LogicalOutput	Address
<b>Awake and Disarm Commands (01h and 11h)</b>	
Partition	EST_Index

### Appendix A.7. Set Date and Time To Panel

The following Map Descriptors are required to set the Date and Time of the EST Panel:

First poke month (1-12), day (0-31) and year (yyyy) at offset 1, 2 and 3 then poke 1 at offset 0 to issue SetDate command.

First poke hour (0-23), min (0-59) and sec (0-59) at offset 1, 2 and 3 then poke 1 at offset 0 to issue SetTime command.

Map_Descriptors						
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_Name	EST_Object_Type	Length
MD_Set_Date	DA_Set_Date	0	WrBx	EST_1	SetDate	1
MD_Set_Time	DA_Set_Time	0	WrBx	EST_1	SetTime	1

The following Map Descriptor will enable the FieldServer to synchronize its clock with the EST panel clock:

Map_Descriptors						
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_Name	EST_Report_Type	Length
MD_SLV_TIME	DA_TIME	0	Passive	EST_1	Date Time	6

## Appendix B. TROUBLESHOOTING

### Appendix B.1. Multiple Com Errors

To solve multiple com errors, try adding an IC\_Timeout parameter to the Client Side Nodes and set it to 0.5s.

**Appendix C. REFERENCE**

**Appendix C.1. EST3 Data Format**

**Appendix C.1.1. Input data format**

Each input device is associated with a 16-bit value. The contents of the 16 bits are used to generate status types as shown in the table below. Each incoming event is allocated to a status type depending on its device type as shown in the table. It is possible to override this default status type allocation. Refer to [Appendix A.5](#) for more information.

Type	Encode Value	Alarm	Trouble	Supervisory	Monitor	CO Active	Others
Smoke	0	Y					
Pull	1	Y					
Heat	2	Y					
Waterflow	3	Y					
Stagetwo	4	Y					
Zone	5	Y					
AlarmVerify	6		Y				
PreAlarm	7		Y				
Security	8		Y				
Station	9						Y
Emergency	10		Y				
Supervisory	11			Y			
Valve	12			Y			
GateValve	13			Y			
Tamper	14			Y			
Temperature	15			Y			
Power	16			Y			
Signal	17			Y			
TroubleOpen	18		Y				
TroubleShort	19		Y				
LocalTrouble	20		Y				
MaintAlert	21		Y				
DirtyHead	22		Y				
DeviceComm	23		Y				
GroundFault	24		Y				
InternalTrouble	25		Y				
BadType	26		Y				
BadPersonality	27		Y				
UnexpectedDevice	28		Y				
CommonTrouble	29		Y				
RelayConfirmation	30						Y
Monitor	31				Y		
LocalMonitor	32				Y		
Damper	33				Y		
Fan	34				Y		
Door	35				Y		

Type	Encode Value	Alarm	Trouble	Supervisory	Monitor	CO Active	Others
Switch	36						Y
FirePhone	37		Y				
TestEvent	38						Y
DeviceCompatabilityFault	39						Y
LogicAnd	40	Y					
LogicMatrix	41	Y					
ServiceGroup	42	Y					
TimeControl	43						Y
GuardPatrol	44		Y				
CheckInGroup	45						Y
<i>Internal use only</i>	46						Y
Disabled	47						Y
<i>Internal use only</i>	48						Y
Instruction Text	49						Y
Object Running	50						Y
Access Trouble	51		Y				
AC Fail	52						Y
Interlockfeedback	53				Y		
FeedbackFailure	54		Y				
CO Alarm	55					Y	
CO Supervisory	56					Y	
CO Monitor	57					Y	
SmokeSupervisory	58			Y			
SensorBypass	59		Y				
End of Life	60		Y				
CO Alarm3	61					Y	
CO Supervisory3	62					Y	
CO Monitor3	63					Y	
Audible	64		Y				
Visual	65		Y				
SupervisedOutput	66		Y				
NonSupervisedOutput	67		Y				
CommonAlarmPutput	68		Y				
CommonTroubleOutput	69		Y				
CommonSupervisoryOutPut	70		Y				
CommonMonitorOutput	71		Y		Y		
LEDOOutput	72						Y
AnalogOutput	73						Y
AudioMessage	74						Y
Amplifier	75						Y
Access Output	76						Y
Interlock	77				Y		
Control Auxil	78				Y		
Genesis Audible Visual	79				Y		
Spare	81-99						Y
DeviceInitFault	100		Y				

Type	Encode Value	Alarm	Trouble	Supervisory	Monitor	CO Active	Others
BaseTypeFault	101		Y				
SensitivityFault	102		Y				
Device Config Fault	103		Y				
Riser Fault	104		Y				
Invlaid Address	105		Y				
Duplicate Fault	106		Y				
Zone Alarm	107	Y					
Zone Supervisory	108			Y			
Zone Monitor	109				Y		
COMBO ALARM	110	Y					
HEAT ACTIVE	111	Y					
Spare	112-126						Y
Card	127		Y				
SecurityAlarm	128						Y
SecurityFault	129						Y
SecurityTamper	130						Y
SecurityMaintenance	131						Y
Spare	132						Y
SecurityDisarmed	133						Y
SecurityExitTimer	134						Y
SecurityEntryTimer	135						Y
SecurityBypassed	136						Y
SecurityStay	137						Y
SecurityAway	138						Y
SecurityStayFail	139						Y
SecurityAwayFail	140						Y
Spare	141						Y
Spare	142						Y
Internal Use only	143						Y
Security Disabled	144						Y
Spare	145-150						Y
CmdList	151						Y

### Appendix C.1.2. Output data format

Each output device is associated with a 16-bit value. This value consists of two byte fields, the command (LSB) and the attribute (MSB).

FieldServer Data Array Object (16-bit)	
Attribute (8-bit)	Command (8-bit)

These values are provided by the EST panel and are described in the EST3 ECP protocol specification as follows. These values will be put in the Command field.

Command	Encode Value
Activate / Away	01h
Enable	02h
Restore / Disarm	11h
Disable	12h

The attribute byte must be set for writes to Relay, LED outputs and DeviceTest.

For Relays, the attribute values are:

Attribute (Relay)	Encode Value
Enable	2h
Activate	101h
Restore	11h
Disable	12h

For LED's, the attribute values are:

Attribute (LED)	Encode Value
On	101h
Slow Flash	201h
Fast Flash	301h
Disable	12h
Enable	2h
Off	1h

For DeviceTest (Signature Devices Only), the attribute values are:

Attribute (DeviceTest)	Encode Value
TestActive1 <i>Places the device in its primary active state (Alarm, Monitor, Supervisory, SecurityAlarm, etc.)</i>	201h
TestTrouble <i>Places the device in CommonTrouble</i>	301h
TestActive2 <i>Places security devices capable of tamper, in SecurityTamper Places sensors capable of PreAlarm in Prealarm</i>	401h



## Appendix C.2. EST3 ECP Connection Statistics

The EST3 ECP protocol has two distinct levels:

- A poll-response connection is maintained by the EST panel acting as the master. The bytes transferred in this process are displayed on the FieldServer as SCADA bytes transmitted and received.
- The FieldServer acts as a client by inserting requests into its response messages; the EST panel then inserts the requested data into its subsequent polling messages. The bytes transferred in this process are displayed on the FieldServer as PLC bytes transmitted and received.

The following statistics are available on the Driver Aspects screen when viewing Connection Details (select the EST Connection in Connection Overview, and press space repeatedly to page to this screen):

Statistic Name	Description
DLL Rx Char	Data Link Layer Bytes Received
DLL Rx Msg	Data Link Layer Messages Received
DLL Tx Char	Data Link Layer Bytes Transmitted
DLL Tx Msg	Data Link Layer Messages Transmitted
DLL Timeouts	Data Link Layer Timeouts
DLL IC Timeouts	Data Link Layer Inter-Character Timeouts
DLL Streaming	Data Link Layer Streaming Errors
DLL Bad Replies	Data Link Layer Bad Replies
DLL Link Control	The driver lost communications with the panel (due to panel reset or power-cycle or cable interruption) and is trying to re-establish communications over the communications link or connection
Gateway Type	ECP Gateway Type I/II/III
Data Rx Char	Data Bytes Received
Data Rx Msg	Data Messages Received
Data Tx Char	Data Bytes Transmitted
Data Tx Msg	Data Messages Transmitted
Conn Rx Char	Total Bytes Received on Connection
Conn Rx Msg	Total Messages Received on Connection
Conn Tx Char	Total Bytes Transmitted on Connection
Conn Tx Msg	Total Messages Transmitted on Connection
Delta Rx Char	Delta Message Bytes Received
Delta Rx Msg	Delta Messages Received

Appendix C.3. Driver Error Messages

Error Message	Description and Action Required
EST3:#01 FYI. For an Old EST3 Panel, Read MD length must be even.	<p>This message will be displayed once if any Map Descriptor in a configuration file has an odd length. Old EST3 Panels (before 2002) simply do not communicate if length is odd.</p> <p>This message can be ignored if it is new EST3 panel. But if it is old or unknown, try changing the Map Descriptor length to an even value before contacting technical support.</p>
EST3:#02 Err. Read MD length must > 0 and <= %d	<p>Numbers of addresses to be read per request are limited.</p> <p>Edit configuration file to set Map Descriptor length within specified range.<sup>8</sup></p>
EST3:#03 Err. Write MD must have length = 1	<p>One Map Descriptor can write only 1 value. Change length to 1.<sup>8</sup></p>
EST3: #04 Unknown Escape Sequence 1B %02X considered as %02X	<p>This message is printed if the FieldServer receives sequences that are not specified. Specified sequences are 1B 82, 1B 8D and 1B 9B. This message is for information only and can be ignored unless the data stored by the FieldServer is invalid in which case, call Tech support.</p>

<sup>8</sup> Upload the configuration file, make the necessary changes, download to the FieldServer and cycle power to the FieldServer for the changes to take effect.