



---

A Sierra Monitor Company

**Driver Manual**  
**(Supplement to the FieldServer Instruction Manual)**

**FS-8700-93 EST QuickStart**

**APPLICABILITY & EFFECTIVITY**

**Effective for all systems manufactured after May 1, 2001**

|                           |             |
|---------------------------|-------------|
| <b>Driver Version:</b>    | <b>1.03</b> |
| <b>Document Revision:</b> | <b>1</b>    |

## TABLE OF CONTENTS

|                                                                         |           |
|-------------------------------------------------------------------------|-----------|
| <b>1. EST QuickStart Description .....</b>                              | <b>3</b>  |
| <b>2. Driver Scope of Supply.....</b>                                   | <b>4</b>  |
| 2.1. Supplied by FieldServer Technologies for this driver .....         | 4         |
| <b>3. Hardware Connections .....</b>                                    | <b>5</b>  |
| 3.1. Hardware Connection Tips / Hints.....                              | 6         |
| <b>4. Configuring the FieldServer as an EST QuickStart Client .....</b> | <b>7</b>  |
| 4.1. Device States .....                                                | 7         |
| 4.2. Data Array Memory Allocation by Card .....                         | 8         |
| 4.3. Device Addressing.....                                             | 10        |
| 4.4. Data Arrays/Descriptors .....                                      | 11        |
| 4.5. Client Side Connection Descriptions .....                          | 12        |
| 4.6. Client Side Node Descriptors .....                                 | 12        |
| 4.7. Client Side Map Descriptors .....                                  | 13        |
| 4.7.1. <i>FieldServer Related Map Descriptor Parameters</i> .....       | 13        |
| 4.7.2. <i>Driver Related Map Descriptor Parameters</i> .....            | 13        |
| 4.8. Map Descriptor Example. ....                                       | 14        |
| <b>Appendix A. Advanced Topics .....</b>                                | <b>15</b> |
| Appendix A.1. FieldServer Synchronization .....                         | 15        |
| A.1.1. Clear All Data Arrays - Map Descriptor Example. ....             | 16        |
| Appendix A.2. Message Types Supported .....                             | 17        |
| <b>Appendix B. Error Messages .....</b>                                 | <b>20</b> |
| Appendix B.1. Driver Statistics .....                                   | 21        |

## 1. EST QuickStart Description

The EST QuickStart Serial driver allows the FieldServer to record data from Edwards Systems Technology QS1-2/4-8 and QSC Fire Panels over the RS-232 printer port. The FieldServer acts as a Passive Client receiving messages and updating its internal Data Arrays - recording the status of devices connected to the EST QuickStart fire alarm control panel. The EST QuickStart panels currently do not support duplexed communications; therefore this driver has only been designed to listen for messages passively and not to respond or to generate polls.

The main purpose of this driver is to record the status of Fire Alarm System points in bit oriented Data Arrays. It is limited by the information that the EST QuickStart control panels broadcasts in the form of text messages through its RS-232 printer port.

The types of EST messages supported by this driver are listed below. Device status to Data Array mapping is also provided. Since there are a variety of optional cards that can be selected for each EST QuickStart installation, each FieldServer application must be specifically configured before use. FieldServer data storage arrays for panel status will be allocated according to the number and type of cards implemented. Message information will also be stored according to card and panel addressing.

This driver is connection oriented; therefore, only one EST Panel may be connected to any single RS-232 FieldServer port.

The driver currently accommodates European and American Conventional/Intelligent panels using the American English protocol with provisions for using the British English protocol at a later date.

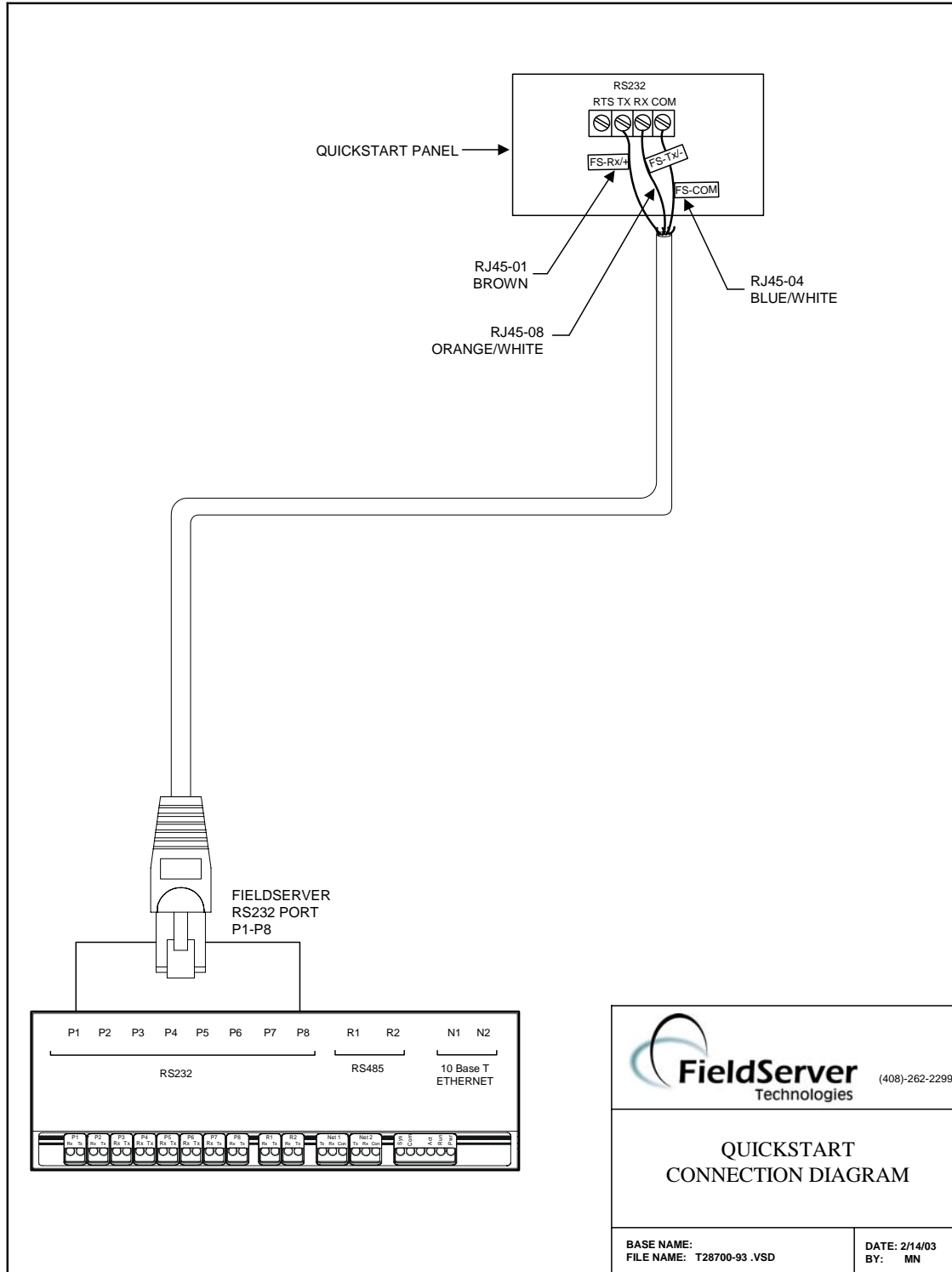
## 2. Driver Scope of Supply

### 2.1. Supplied by FieldServer Technologies for this driver

| FieldServer Technologies PART # | Description           |
|---------------------------------|-----------------------|
| FS-8917-16                      | UTP to Terminal cable |
| FS-8700-93 EST3                 | Driver Manual.        |

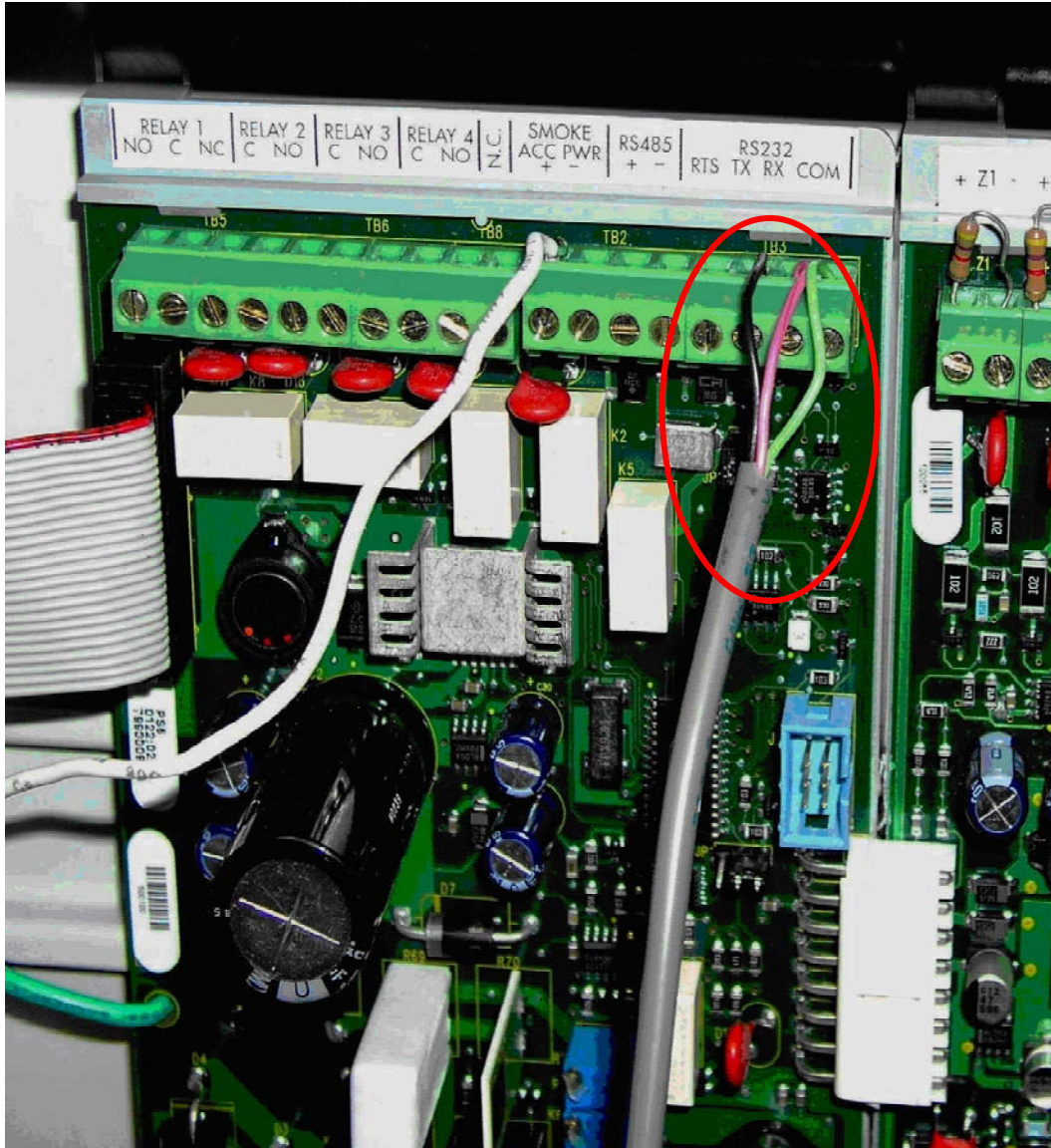
### 3. Hardware Connections

The FieldServer is connected to the EST QuickStart power supply card as shown below. Configure the EST QuickStart panel according to manufacturer's instructions.



### 3.1. Hardware Connection Tips / Hints

The QuickStart panels must be pre-configured using their own proprietary configuration tool to send messages to the RS-232 printer port. Ensure that this is done before attempting to connect to the FieldServer and verify that messages are being sent using a utility like Hyperterminal.



## 4. Configuring the FieldServer as an EST QuickStart Client

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

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with a EST QuickStart Server.

It is possible to connect the Edward QuickStart panel to any RS-232 or RS-485 (with adapter) port on a FieldServer. These ports need to be configured for Protocol="qstart" in the configuration files.

### 4.1. Device States

Each addressable device connected to the QuickStart panel (real or pseudo) can have multiple states: alarm supervisory, monitor, other (security), and troubles. These states are not exclusive; therefore, a single addressable point may occupy multiple states at the same time.

Although there may be several trouble states for each device, there can only be a single alarm, supervisory, monitor and other (security) state per address. Therefore any restoring messages for these four states indicate that the device is clear of any active problems for these types of conditions.

There may, however, be several trouble states for a single addressable port. These trouble states will be recorded in a separate trouble array of 2 byte elements. Each array element will correspond to a device address and each bit within the element will correspond to a trouble type. The mapping of message trouble types to element bit configuration is given below. For modeling simplicity and flexibility, it is assumed that all addressable points (real or pseudo) can exist in all states.

If the number of trouble states increases in the future, 4 byte element arrays can be substituted for the 2 byte element arrays currently used with some additional driver code modifications.

| Message Type                 | State | Bit /UINT16/Packed_Bit |
|------------------------------|-------|------------------------|
| TROUBLE OPEN<br>TBL OPEN RST | T     | 0                      |
| TROUBLE SHRT<br>TBL SHRT RST | T     | 1                      |
| LCL TROUBLE<br>LCL TRBL RST  | T     | 2                      |
| DIRTY HEAD<br>DTY HEAD RST   | T     | 3                      |
| COMMN FAULT<br>COMM FLT RST  | T     | 4                      |
| GROUND FAULT<br>GND FLT RST  | T     | 5                      |
| INTERNAL TBL<br>INTR TRBL RS | T     | 6                      |
| BAD TYPE<br>BAD TYPE RST     | T     | 7                      |
| BAD PRSONATY<br>BAD PRSTY RS | T     | 8                      |
| UNEPECT DEV<br>UNEC DEV RST  | T     | 9                      |
| TROUBLE<br>TROUBLE RST       | T     | 10                     |
| TEST<br>TEST RST             | T     | 11                     |
| DEV COMPATBL<br>DEV COMP RST | T     | 12                     |
| SERVICE GRP<br>SERVICE G RS  | T     | 13                     |
| DISABLED<br>DISABLED RST     | T     | 14                     |
| DISAB SOUND<br>DISSOUND RST  | T     | 15                     |

#### 4.2. Data Array Memory Allocation by Card

Each QuickStart card (real and pseudo) is configured with BIT and UNIT16 Data Arrays. The following states are recorded in the BIT array.

- Alarms
- Supervisory
- Monitor
- Other (security)

Alarms will be stored in the first quarter of the array; Supervisory conditions will be stored in the second quarter etc. according to the above ordering.

Troubles are recorded in a separate array of UINT16 or Packed\_Bit elements described above. Each element represents a uniquely addressable point, and each bit within the

element represents a different type of trouble condition which is described by the specific trouble message.

The following table lists the BIT and UINT16/Packed\_Bit Data Array sizes that should be configured for every panel by default, and arrays for each optionally installed card. The Data Arrays configured for all QuickStart panels are for cards (real and pseudo) which always have fixed card addresses. Not all of these default Data Arrays will be utilized for every installation. Optionally installed cards can have variable card id #'s depending on the sequence the cards are installed.

Data Arrays are referenced relatively according to their declared size. Therefore, care must be taken to ensure that the array sizes are configured as below.

| <b>Card/Array Descriptions</b>  | <b>Array Size</b> | <b>Data Type</b>  |
|---------------------------------|-------------------|-------------------|
| Pseudo points                   | 200               | BIT               |
| Pseudo point troubles           | 50                | UINT16/Packed_Bit |
| Zones                           | 2000              | BIT               |
| Zone troubles                   | 500               | UINT16/Packed_Bit |
| AND group                       | 400               | BIT               |
| AND group troubles              | 100               | UINT16/Packed_Bit |
| Matrix group                    | 400               | BIT               |
| Matrix group troubles           | 100               | UINT16/Packed_Bit |
| Service group                   | 800               | BIT               |
| Service group troubles          | 200               | UINT16/Packed_Bit |
| Time control                    | 200               | BIT               |
| Time control troubles           | 50                | UINT16/Packed_Bit |
| Output group                    | 2000              | BIT               |
| Output group troubles           | 500               | UINT16/Packed_Bit |
| CPU reserved                    | 200               | BIT               |
| CPU reserved troubles           | 50                | UINT16/Packed_Bit |
| Power card                      | 200               | BIT               |
| Power card troubles             | 50                | UINT16/Packed_Bit |
| <b>Optional Cards</b>           |                   |                   |
| ADC (dual loop)                 | 2000              | BIT               |
| ADC (dual loop) troubles        | 500               | UINT16/Packed_Bit |
| SLIC                            | 2000              | BIT               |
| SLIC troubles                   | 500               | UINT16/Packed_Bit |
| Class A card                    | 200               | BIT               |
| Class A card troubles           | 50                | UINT16/Packed_Bit |
| Class B card                    | 200               | BIT               |
| Class B card troubles           | 50                | UINT16/Packed_Bit |
| Relay card                      | 200               | BIT               |
| Relay card troubles             | 50                | UINT16/Packed_Bit |
| Annunciator (per card)          | 200               | BIT               |
| Annunciator troubles (per card) | 50                | UINT16/Packed_Bit |
| Dialer                          | 200               | BIT               |
| Dialer troubles                 | 50                | UINT16/Packed_Bit |

### 4.3. Device Addressing

The following table defines the address space expected by the driver for each type of card. The table includes both physical and pseudo points used by the QuickStart systems.

|                  | Panel | Card      | Address     | Card Type        |
|------------------|-------|-----------|-------------|------------------|
| Point Addressing | 0     | 0         | 0-18 (49)   | pseudo points    |
|                  | 0     | 17        | 0-480 (499) | zone             |
|                  | 0     | 18        | 0-64 (99)   | AND group        |
|                  | 0     | 19        | 0-64 (99)   | matrix group     |
|                  | 0     | 20        | 0-128 (199) | service group    |
|                  | 0     | 21        | 0-32 (49)   | time control     |
|                  | 0     | 22        | 0-250 (499) | output group     |
|                  | 1     | 0         | 0-32 (49)   | CPU (reserved)   |
|                  | 2     | 0         | 0-33 (49)   | annunciator      |
|                  | 3     | 0         | 0-33 (49)   | annunciator      |
|                  | 4     | 0         | 0-33 (49)   | annunciator      |
|                  | 5     | 0         | 0-33 (49)   | annunciator      |
|                  | 6     | 0         | 0-33 (49)   | annunciator      |
|                  | 7     | 0         | 0-33 (49)   | annunciator      |
|                  | 8     | 0         | 0-33 (49)   | annunciator      |
|                  | 9     | 0         | 0-33 (49)   | annunciator      |
|                  | 1     | 01-07     | 0-341 (499) | ADC              |
|                  | 1     | 01-07     | 0-275 (499) | SLIC             |
|                  | 1     | 01-13     | 0-19 (49)   | Class B (ZA16-8) |
|                  | 1     | 01-13     | 0-11 (49)   | Class A (ZA8-2)  |
|                  | 1     | 01-13     | 0-11 (49)   | Relay (ZR-8)     |
|                  | 1     | 14        | 0-16 (49)   | Dialer (DLD)     |
| 1                | 15    | 0-24 (49) | Power (PS6) |                  |

#### 4.4. Data Arrays/Descriptors

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

| 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.                                                                          | Bit, UInt16, Packed_Bit          |
| Data_Array_Length | Number of Data Objects. Must be larger than the data storage area required by the Map Descriptors for the data being placed in this array. | 1-10,000                         |

#### Example

```
// Data Arrays

Data_Arrays
Data_Array_Name,      Data_Format,      Data_Array_Length,
DA_AI_01,             UInt16,           200
DA_AO_01,             UInt16,           200
DA_DI_01,             Bit,              200
DA_DO_01,             Bit,              200
CLEAR_FLAG            Bit,              16
```

#### 4.5. Client Side Connection Descriptions

| Section Title |                                                               |                           |
|---------------|---------------------------------------------------------------|---------------------------|
| Connections   |                                                               |                           |
| Column Title  | Function                                                      | Legal Values              |
| Port          | Specify which port the device is connected to the FieldServer | P1-P8, R1-R2 <sup>1</sup> |
| Protocol      | Specify protocol used                                         | qstart                    |
| Baud*         | Specify baud rate                                             | 9600 (vendor limitation)  |
| Parity*       | Specify parity                                                | <b>None</b>               |
| Data_Bits*    | Specify data bits                                             | <b>8</b>                  |
| Stop_Bits*    | Specify stop bits                                             | <b>1</b>                  |
| Handshaking*  | Specify hardware handshaking                                  | <b>None</b>               |
| Ic_timeout    | Specify intercharacter timeout                                | <b>2.0s</b>               |

#### Example

|             |                         |       |         |              |            |  |
|-------------|-------------------------|-------|---------|--------------|------------|--|
| //          | Client Side Connections |       |         |              |            |  |
| Connections |                         |       |         |              |            |  |
| Port,       | Protocol,               | Baud, | Parity, | Handshaking, | ic_timeout |  |
| P3,         | qstart,                 | 9600, | None,   | None,        | 2.0s       |  |

#### 4.6. Client Side Node Descriptors

| Section Title |                                                               |                                  |
|---------------|---------------------------------------------------------------|----------------------------------|
| Nodes         |                                                               |                                  |
| Column Title  | Function                                                      | Legal Values                     |
| Node_Name     | Provide name for node                                         | Up to 32 alphanumeric characters |
| Node_ID       | Address of physical Server node                               | 1-255                            |
| Protocol      | Specify protocol used                                         | qstart                           |
| Connection    | Specify which port the device is connected to the FieldServer | P1-P8, R1-R2 <sup>1</sup>        |

#### Example

|            |                   |           |            |  |
|------------|-------------------|-----------|------------|--|
| //         | Client Side Nodes |           |            |  |
| Nodes      |                   |           |            |  |
| Node_Name, | Node_ID,          | Protocol, | Connection |  |
| Node_A,    | 1,                | qstart,   | P8         |  |

<sup>1</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.

## 4.7. Client Side Map Descriptors

### 4.7.1. FieldServer Related Map Descriptor Parameters

| Section Title       |                                                                  |                                                             |
|---------------------|------------------------------------------------------------------|-------------------------------------------------------------|
| Map Descriptors     |                                                                  |                                                             |
| 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                                | Server                                                      |

### 4.7.2. Driver Related Map Descriptor Parameters

| Section Title   |                                                                                           |                                                                   |
|-----------------|-------------------------------------------------------------------------------------------|-------------------------------------------------------------------|
| Map Descriptors |                                                                                           |                                                                   |
| 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 |
| Msg_Type        | Identifies the Map Descriptor as being for a trouble message or any other type of message | trouble, other                                                    |
| Panel           | Id of QuickStart panel                                                                    | 0-22                                                              |
| Card            | Id of Card                                                                                | 0-9                                                               |
| Length          | Length of Map Descriptor                                                                  | Permissible range of addresses based on Data Array size           |
| Address         | Starting address of read block                                                            | 0 for passive Clients                                             |

### 4.8. Map Descriptor Example.

This is a typical example of a Passive Client Map Descriptor, which only reads incoming messages, and does not poll or respond to messages. All messages will be recorded into the Data Arrays designated to this port. The Data Arrays must be of the size designated in Data Arrays above. The Message Type, Panel, Card, and Address fields are used to match the appropriate Map Descriptor for updating data.

```
// Client Side Map Descriptors
```

| Map Descriptor Name | Data Array Name  | Data Array Offset | Function | Node name | Msg_type | Panel | Card | Address | Length |
|---------------------|------------------|-------------------|----------|-----------|----------|-------|------|---------|--------|
| SLV_PSEUDO,         | DA_PSEUDO,       | 0,                | Server,  | Node_A ,  | Other,   | 0,    | 0,   | 0,      | 50     |
| SLV_PSEUDO_TR,      | DA_PSEUDO_TR,    | 0,                | Server,  | Node_A ,  | Trouble, | 0,    | 0,   | 0,      | 50     |
| SLV_ZONE,           | DA_ZONE,         | 0,                | Server,  | Node_A ,  | Other,   | 0,    | 17,  | 0,      | 500    |
| SLV_ZONE_TR,        | DA_ZONE_TR,      | 0,                | Server,  | Node_A ,  | Trouble, | 0,    | 17,  | 0,      | 500    |
| SLV_AND_GRP,        | DA_AND_GRP,      | 0,                | Server,  | Node_A ,  | Other,   | 0,    | 18,  | 0,      | 100    |
| SLV_AND_GRP_TR,     | DA_AND_GRP_TR,   | 0,                | Server,  | Node_A ,  | Trouble, | 0,    | 18,  | 0,      | 100    |
| SLV_MTRX_GRP,       | DA_MTRX_GRP,     | 0,                | Server,  | Node_A ,  | Other,   | 0,    | 19,  | 0,      | 100    |
| SLV_MTRX_GRP_TR,    | DA_MTRX_GRP_TR,  | 0,                | Server,  | Node_A ,  | Trouble, | 0,    | 19,  | 0,      | 100    |
| SLV_SVC_GRP,        | DA_SVC_GRP,      | 0,                | Server,  | Node_A ,  | Other,   | 0,    | 20,  | 0,      | 200    |
| SLV_SVC_GRP_TR,     | DA_SVC_GRP_TR,   | 0,                | Server,  | Node_A ,  | Trouble, | 0,    | 20,  | 0,      | 200    |
| SLV_TIME_CTRL,      | DA_TIME_CTRL,    | 0,                | Server,  | Node_A ,  | Other,   | 0,    | 21,  | 0,      | 50     |
| SLV_TIME_CTRL_TR,   | DA_TIME_CTRL_TR, | 0,                | Server,  | Node_A ,  | Trouble, | 0,    | 21,  | 0,      | 50     |
| SLV_OP_GRP,         | DA_OP_GRP,       | 0,                | Server,  | Node_A ,  | Other,   | 0,    | 22,  | 0,      | 500    |
| SLV_OP_GRP_TR,      | DA_OP_GRP_TR,    | 0,                | Server,  | Node_A ,  | Trouble, | 0,    | 22,  | 0,      | 500    |
| SLV_CPU_RSRV,       | DA_CPU_RSRV,     | 0,                | Server,  | Node_A ,  | Other,   | 1,    | 0,   | 0,      | 50     |
| SLV_CPU_RSRV_TR,    | DA_CPU_RSRV_TR,  | 0,                | Server,  | Node_A ,  | Trouble, | 1,    | 0,   | 0,      | 50     |

The driver internally calculates the correct offset for each message type. See the section on Data Array Memory Allocation for each card above.

QuickStart is a passive Client and neither reads or writes to another device. Server is synonymous with a passive Map Descriptor waiting for message to store data.

There can only be one node per port. This field indirectly links the FieldServer port to a data storage array. See description of Node definition above.

Besides using panel id, card id, and device address to identify a responsible Map Descriptor and destination Data Array - messages are divided as either troubles or other types.

For this driver, address should always be zero. It defines the lowest possible expected address value.

Acceptable ranges for address are defined by the card type. Each card has an expected maximum address which this field defines.

---

**Appendix A. Advanced Topics**

---

**Appendix A.1. FieldServer Synchronization**

To synchronize the FieldServer's Data Arrays with the QuickStart control panel's status, the panel must have no active or disabled points on FieldServer startup. Alternatively, a panel can be hard/soft restarted to clear all point status, and then re-annunciate any active devices after the FieldServer has been started. This should only be undertaken by qualified technicians.

Should the panel be soft restarted unexpectedly, the FieldServer will recognize the "restart" command and clear all internal Data Arrays. There will be no need for FieldServer restart. For a hard restart, the FieldServer Data Arrays should be manually cleared by changing a bit in the CLEAR\_FLAG Data Array through RUINET, then soft restarting the panel again to ensure that status of all points is re-annunciated and recorded by the FieldServer.

A method for manually clearing all Data Arrays for the QuickStart driver is available. This is useful if the QuickStart Panel is hard re-started, and the user wishes to synchronize the FieldServer Data Arrays. To enable this feature, a CLEAR\_FLAG Data Array needs to be declared in the configuration file that is 16 bits long. To invoke a manual clear, any bit within the array needs to be set to high or "1". (See FieldServers RUINET user manual for detailed instructions on user operations) After the clearing function has been completed, all bits within the CLEAR\_FLAG array should be automatically set to zero again.

**A.1.1. Clear All Data Arrays - Map Descriptor Example.**

```
// Data Arrays
Data_Arrays
Data_Array_Name, Data_Format, Data_Array_Length
CLEAR_FLAG      Bit,          16
```

| Map_Descriptor_Name, | Data_Array_Name, | Data_Array_Offset, | Function, | Node_name, | Panel, | Card, | Address, | Length |
|----------------------|------------------|--------------------|-----------|------------|--------|-------|----------|--------|
| CLEAR_CHK,           | CLEAR_FLAG,      | 0,                 | Wrbx,     | Node_A,    | 0,     | 0,    | 0,       | 16     |

This array needs to be declared in order to manually reset all Data Arrays

QuickStart is a passive Client and neither reads or writes to another device. WRBX monitors the identified Data Array for changes, and invokes the clear function when this occurs.

There can only be one node per port. This field indirectly links the FieldServer port to a data storage array. See description of Node definition above.

Use these default panel id, card id, and device address as dummy values

Set to the defined length of the CLEAR\_FLAG array

## Appendix A.2. Message Types Supported

The primary purpose of this driver is to record the status of devices connected to the QuickStart Fire Control Panel by interpreting the text messages sent to its printer port. Not all messages will be interpreted, as many messages and reports do not directly affect device status. The following set of events and messages will be recognized and recorded in the Data Arrays.

| Event Description Text Strings | State | Comments                                      |
|--------------------------------|-------|-----------------------------------------------|
| ACTIVATE RESTART               |       | -OPERATOR COMMAND- clears all the Data Arrays |
| QUIESCENT<br>QUIE RST          |       | Not Implemented                               |
| ALARM ACTIVE<br>ALARM RST      | A     |                                               |
| PULL STATION<br>PULL STA RST   | A     |                                               |
| HEAT ALARM<br>HEAT ALM RS      | A     |                                               |
| WATERFLOW<br>WATERFLOW RS      | A     |                                               |
| STAGE ONE<br>STAGE ONE RS      | A     |                                               |
| SUPERVISORY<br>SUP RST         | S     |                                               |
| TAMPER<br>TAMPER RST           | S     |                                               |
| SECURITY<br>SECURITY RST       | O     |                                               |
| FIREPHONE<br>PHONE RST         | M     |                                               |
| MONITOR<br>MONITOR RST         | M     |                                               |
| LATCH SUPV<br>LAT SUPV RST     | S     |                                               |
| LATCH TAMPER<br>LAT TAMP RST   | S     |                                               |
| UNKNOWN<br>UNKNOWN RST         | O     |                                               |
| ALARM VERIFY<br>ALM VFY RST    | M     |                                               |
| PREALARM<br>PREALARM RST       | M     |                                               |
| TROUBLE OPEN<br>TBL OPEN RST   | T     |                                               |
| TROUBLE SHRT<br>TBL SHRT RST   | T     |                                               |
| LCL TROUBLE<br>LCL TRBL RST    | T     |                                               |
| MAINT ALERT<br>MAT ALERT RS    | M     |                                               |
| DIRTY HEAD<br>DTY HEAD RST     | T     |                                               |
| COMMN FAULT<br>COMM FLT RST    | T     |                                               |

| Event Description Text Strings | State | Comments |
|--------------------------------|-------|----------|
| GROUND FAULT<br>GND FLT RST    | T     |          |
| INTERNAL TBL<br>INTR TRBL RS   | T     |          |
| BAD TYPE<br>BAD TYPE RST       | T     |          |
| BAD PRSONATY<br>BAD PRSTY RS   | T     |          |
| UNEPECT DEV<br>UNEC DEV RST    | T     |          |
| TROUBLE<br>TROUBLE RST         | T     |          |
| RELAY CONFIR<br>RLY CFRM RST   | M     |          |
| LCL MONITOR<br>LCL MNTR RST    | M     |          |
| SWITCH<br>SWITCH RST           | M     |          |
| TEST<br>TEST RST               | T     |          |
| DEV COMPATBL<br>DEV COMP RST   | T     |          |
| AND GROUP<br>AND GRP RST       | A     |          |
| MATRIX GROUP<br>MATRIX G RST   | A     |          |
| SERVICE GRP<br>SERVICE G RS    | T     |          |
| TIME CONTROL<br>TIME CNTL RS   | M     |          |
| ACK<br>ACK RST                 | M     |          |
| DISABLED<br>DISABLED RST       | T     |          |
| DISAB SOUND<br>DISSOUND RST    | T     |          |
| OBJECT RUN<br>OBJETRUN RST     | M     |          |
| ZONE ALARM<br>ZONE ALM RST     | A     |          |
| ZONE SUPER<br>ZONE SUP RST     | S     |          |
| ZONE MONITOR<br>ZONE MON RST   | M     |          |
| OUTPUT GROUP<br>OUTPUT G RST   | M     |          |
| AUDIBLE<br>AUDIBLE RST         | M     |          |
| VISUAL<br>VISUAL RST           | M     |          |
| SUPER OUTPUT<br>SUPER OUT RT   | M     |          |
| NONSUPER OUT<br>NONSUP OUT R   | M     |          |

| Event Description Text Strings | State | Comments |
|--------------------------------|-------|----------|
| COMM ALM OUT<br>COMMALOUTRST   | M     |          |
| LED OUTPUT<br>LEDOUTPU RST     | M     |          |

## Appendix B. Error Messages

| Error Message                                                                                                          | Corrective Action                                                                                                                            |
|------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| Qstart:#1 Err. Illegal Map Descriptor length - defaulting to 1                                                         | Check configuration file settings, the length field is not appropriately set. <sup>2</sup>                                                   |
| Qstart:#2 Err. Card value error                                                                                        | Check configuration file settings, printed circuit board card id numbers must be 0-22 per QuickStart's addressing scheme above. <sup>2</sup> |
| Qstart:#3 Err. Panel value error                                                                                       | Check configuration file settings, panel id numbers must be 0-9 per QuickStart's addressing scheme above. <sup>2</sup>                       |
| Qstart:#4 Err. Message Type error                                                                                      | Check configuration file settings, message type must be either "trouble", or "other". <sup>2</sup>                                           |
| Qstart:#5 Err. Test file <%s> not found.                                                                               | For developmental diagnostics only, call Technical Support                                                                                   |
| Qstart:#6 Err. Diagnostic line ignored.                                                                                | For developmental diagnostics only, call Technical Support                                                                                   |
| Qstart:#7 Err. parsing Event message\n%s                                                                               | Call Technical Support                                                                                                                       |
| Qstart:#8: Err. storing Event message\n%s                                                                              | Call Technical Support                                                                                                                       |
| Qstart:#9 Err. Message address <%d> exceed Data Array size <%d>                                                        | Call Technical Support                                                                                                                       |
| Qstart:#10 Err. Incoming data is being abandoned on port %d. MapDesc's are required to define storage for message:\n%s | Check that a valid Map Descriptor exists in the configuration files for message. Call Technical Support                                      |
| Qstart:#17 Err. No polling allowed. Presumed write thru abandoned! Map Descriptor <%s>                                 | Writes are not possible; avoid writes from SCADA system or from Ruinet.                                                                      |

<sup>2</sup> Modify the configuration CSV file, transfer the modified file to the FieldServer and reset the FieldServer for the modifications to take effect

### Appendix B.1. Driver Statistics

The following table identifies statistics generated by the EST QuickStart serial driver and their meanings.

| Driver Statistics Recorded | Explanation                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Message received           | Total number of messages of all types                                                                                                                                                                                                                                                                                                                        |
| Bytes received             | Total number of bytes received by all message types.                                                                                                                                                                                                                                                                                                         |
| Message ignored            | Ignored messages include those discarded by the complete function's sliding window criteria, and unrecognized messages in the parsing function. The sliding window discards the first line of three line messages if the correct line pattern is not recognized. Therefore, this stat counts both discarded lines, and messages if they could not be parsed. |
| Protocol                   | Protocol error, the message is recognized but does not take a valid form.                                                                                                                                                                                                                                                                                    |

THIS PAGE INTENTIONALLY LEFT BLANK