

Driver Manual

FS-8700-59 Canatal Satchnet Driver

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after March 2021.



Driver Revision: 1.02
Document Revision: 3.A



fieldserver

MSA Safety
1991 Tarob Court
Milpitas, CA 95035
Website: www.MSAsafety.com

U.S. Support Information:
+1 408 964-4443
+1 800 727-4377
Email: smc-support@msasafety.com

EMEA Support Information:
+31 33 808 0590
Email: smc-support.emea@msasafety.com

Contents

| | | |
|----------|---|-----------|
| 1 | Description | 4 |
| 2 | Driver Scope of Supply | 4 |
| | 2.1 Supplied by MSA Safety..... | 4 |
| | 2.2 Provided by the Supplier of 3 rd Party Equipment | 4 |
| 3 | Hardware Connections | 5 |
| | 3.1 Connection Notes | 6 |
| | 3.1.1 Interconnecting Cable Characteristics | 6 |
| 4 | Data Array Parameters | 7 |
| 5 | Client Side Configuration | 8 |
| | 5.1 Client Side Connection Parameters | 8 |
| | 5.2 Client Side Node Parameters | 9 |
| | 5.3 Client Side Map Descriptor Parameters | 10 |
| | 5.3.1 FieldServer Specific Map Descriptor Parameters | 10 |
| | 5.3.2 Driver Related Map Descriptor Parameters | 10 |
| | 5.3.3 Timing Parameters | 10 |
| | 5.4 Map Descriptor Example | 11 |
| 6 | Server Side Configuration | 12 |
| | 6.1 Server Side Connection Parameters | 12 |
| | 6.2 Server Side Node Parameters..... | 12 |
| | 6.3 Server Side Map Descriptor Parameters..... | 13 |
| | 6.3.1 FieldServer Specific Map Descriptor Parameters | 13 |
| | 6.3.2 Driver Specific Map Descriptor Parameters | 13 |
| | 6.3.1 Timing Parameters | 13 |
| | 6.4 Map Descriptor Example | 14 |
| 7 | Useful Features | 15 |
| | 7.1 Retrieving Data from Tables with Mixed Data Formats..... | 15 |
| | 7.1.1 Map Descriptor Example..... | 15 |
| 8 | Troubleshooting | 16 |
| | 8.1 Improving Communication on Legacy Canatal Devices..... | 16 |
| | 8.2 Driver Error Messages..... | 16 |
| 9 | Reference | 18 |
| | 9.1 Canatal Device Data Tables..... | 18 |
| | 9.2 Recommended Data Array Formats Based on Canatal Data Types | 19 |

1 Description

The Canatal Satchnet Protocol Driver allows the FieldServer to transfer data to and from devices over either RS-232 or RS-485 using Canatal Satchnet Protocol Driver protocol. The FieldServer can emulate either a Server or Client.

This driver provides support for the Canatal International implementation of the “Satchwell Network (Full) and Net2 (Reduced)” protocol.

The Canatal Satchnet Protocol Driver can act as a Satchnet Master.

- Any Satchnet device on any serial port (P1-P2, R1-R2) can be polled.
- Data from Series 2, 4, 5, 6,8, 9 devices can be polled.
- Data from any 'Table' can be read.
- Data in 'Tables' with “write permission” can be written to by the driver.
- The driver can read/write Bit, Byte, Word, BCD and Float Data.

The Canatal Satchnet Protocol Driver can act as a (passive) Server.

- The driver can be polled by a Satchnet Master and return data from the FieldServer's data arrays.
- The driver can emulate Series 2,4,5,6,8,9 devices.
- All “Tables” can be read/written in accordance with their read/write capability.

| FieldServer Mode | Nodes | Comments |
|------------------|-------|---|
| Client | 63 | Expansion above 32 nodes normally requires a repeater |
| Server | 63 | Expansion above 32 nodes normally requires a repeater |

2 Driver Scope of Supply

2.1 Supplied by MSA Safety

| Part # | Description |
|--------|---------------|
| | Driver Manual |

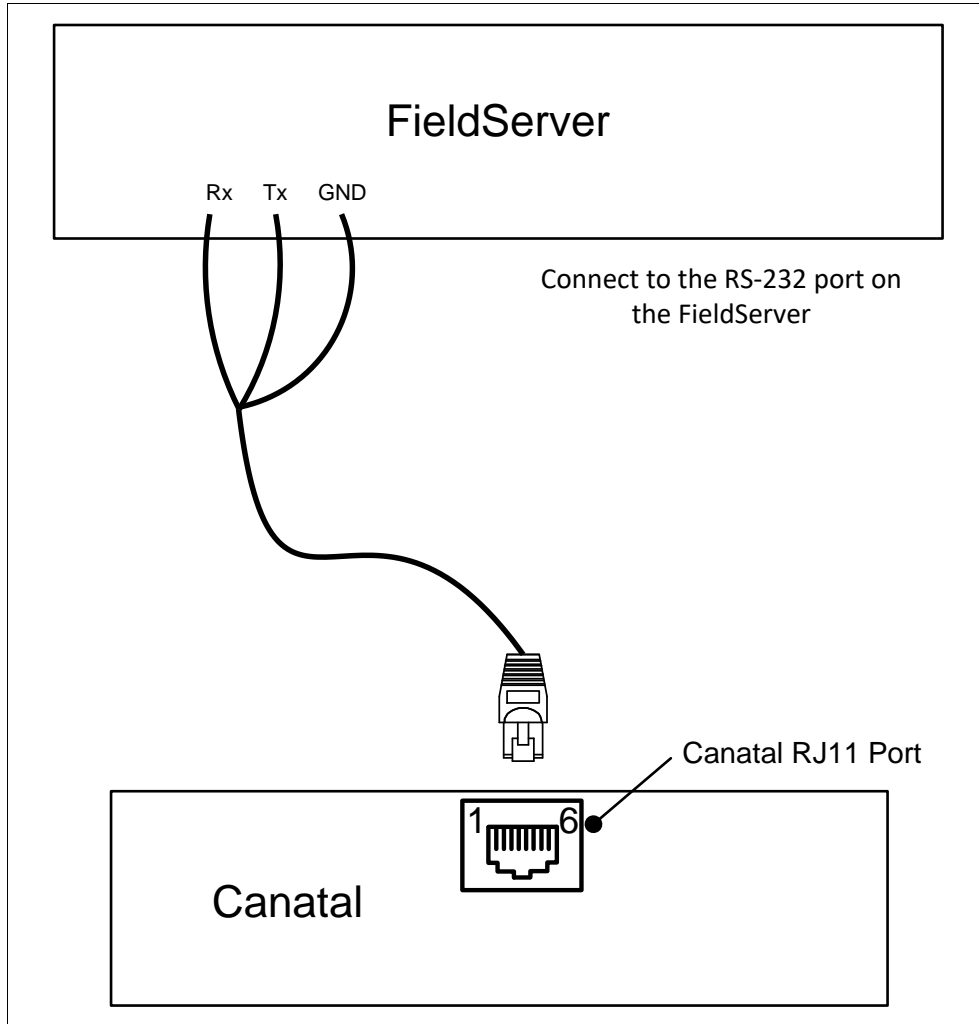
2.2 Provided by the Supplier of 3rd Party Equipment

| Part # | Description |
|--------|-----------------------------------|
| 23068 | Assembly Harness – Canatal Driver |

3 Hardware Connections

The FieldServer is connected to the Canatal Driver as shown below.

Configure the Canatal Driver according to manufacturer's instructions.



3.1 Connection Notes

Set Jumper 16, 17 and 18 on the board to position 2-3.

3.1.1 Interconnecting Cable Characteristics

The interconnecting cable may be composed of twisted or untwisted pair (flat cable) possessing the characteristics below.

- **Conductor Size:** The interconnecting cable shall be composed of two wires of a 24 AWG or larger conductor for solid or stranded copper wires, or for non-copper conductors with sufficient size to yield a DC wire resistance not to exceed 30 ohms per 1000 feet per conductor.
- **Mutual Pair Capacitance:** The capacitance between the two wires in the wire pair shall not exceed 20 picofarads per foot and the value shall be reasonably uniform over the length of the cable.
- **Stray Capacitance:** The capacitance between wires in the cable sheath, with all wires connected to ground shall not exceed 40 picofarads per foot and shall be reasonably uniform over the length of the cable.
- **Pair-to-Pair Balanced Crosstalk:** The balanced crosstalk from one wire pair to any other pair in the same cable sheath shall have a minimum value of 40 decibels of attenuation measured at 150 kilohertz.

An interconnecting cable meeting these specifications will result in a transmission line with nominal characteristic impedance in the order of 100 ohms to frequencies greater than 100 kilohertz and a DC series loop resistance not exceeding 240 ohms.

4 Data Array Parameters

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

| Section Title | | |
|-------------------|--|---|
| Data_Arrays | | |
| Column Title | Function | Legal Values |
| Data_Array_Name | Provide name for Data Array. | Up to 15 alphanumeric characters |
| Data_Array_Format | Provide data format. Each Data Array can only take on one format. | Float, Bit, Byte, UInt16, UInt32, Sint16, Sint32; Refer to Section 9.2 |
| 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. | 1-10000 |

Example

```
// Data Arrays
Data_Arrays
Data_Array_Name , Data_Array_Format , Data_Array_Length
DA_AI_01 , UInt16, , 200
DA_AO_01 , UInt16 , , 200
DA_DI_01 , Bit , , 200
DA_DO_01 , Bit , , 200
```

5 Client Side Configuration

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 a Canatal Satchnet Client.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for Canatal Satchnet Protocol Driver 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 following tables, * indicates an optional parameter and bold legal values are 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 |
| Baud | Specify baud rate. | 1200 , 2400, 4800, 9600 (Vendor limitation) |
| Parity | Specify parity. | Even |
| Data_Bits | Specify data bits. | 7 |
| Stop_Bits* | Specify stop bits. | 1 |
| Protocol | Specify protocol used. | Canatal, Satchnet |
| Poll Delay* | Time between internal polls. | 0-32000 seconds, 1 s |

Example

```
// Client Side Connections
Connections
Port          , Baud  , Parity  , Protocol  , Data_Bits  , Stop_Bits  Port
R1            , 1200 , Even   , Canatal  , 7          , 1          R1
```


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 | Address of physical server node (Canatal device address). | 1-63 |
| Protocol | Specify protocol used. | Canatal, Satchnet |
| Port | Specify FieldServer port the device is connected to. | P1-P2, R1-R2 |
| Cana_Node_Protocol_Type* | Specify which specific protocol is supported by the Canatal device. Series 2, 3, 4 and 5 support only "Network". Series 6, 8 and 9 support both "Network" and "Net2". Specifying this as Net2 will significantly improve communication. | Network, Net2 |

Example

```
// Client Side Nodes
Nodes
Node_Name , Node_ID , Protocol , Port , Cana_Node_Protocol_Type
Conditioner1 , 1 , Canatal , R1 , Net2
```

5.3 Client Side Map Descriptor Parameters

5.3.1 FieldServer Specific Map Descriptor Parameters

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

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 Section 5.2 |
| Length | Length of Map Descriptor. | 1 - 1000 (Maximum value is dependent on the values of Cana_Series and Cana_Table); Refer to Section 9.1 |
| Address | The offset into the Canatal device data table. Each table first element is addressed as zero, the 2nd element as 1 etc. | 0,1,2,3 ...; The address & length must not exceed the maximum length of the table being addressed. |
| Cana_Series | Canatal Corporation Inc. Device Series Number. | 2,4,5,6,9 |
| Cana_Table | The data table in the device to be read. | Non zero integers. Refer to Section 9.1 |
| Cana_Format* | For tables with mixed data formats, the driver assigns a default data format. This default can be overridden using this parameter. Refer to Section 9.1 . | Byte, BCD, BCD4, Float, Word, Percent, Swap_int. |

5.3.3 Timing Parameters

| Column Title | Function | Legal Values |
|---------------|------------------------------|--------------|
| Scan_Interval | Rate at which data is polled | >0.1s |

5.4 Map Descriptor Example

In this example two Map Descriptors are provided. The first reads Device 11 (a Series 2 device) and obtains the Alarm Table. The second Map Descriptor reads Device 23 (a series 9 device) and reads the Cooling analog output (2nd element of table 10).

| Map_Descriptor_Name | Data_Array_Name | Data_Array_Offset | Function | Node_Name | Address | Length |
|---------------------|-----------------|-------------------|----------|-----------|---------|--------|
| DEV11_ALMS | ALARMS | 220 | Rdbc | DEV11 | 0 | 1 |
| DEV23_COOL | ANA_OUT | 0 | Rdbc | DEV23 | 1 | 1 |

| Cana_Series | Cana_Table | Scan_Interval |
|-------------|------------|---------------|
| 2 | 11 | 1.0s |
| 9 | 10 | 1.0s |

In the above example:

- Map_Descriptor_Name – Use a meaningful name. The driver validates Map Descriptors and reports errors referring to the Map Descriptor name.
- Data_Array_Name – The data arrays are defined in the Data_Arrays part of the CSV. Based on these examples the array called ALARMS should be a byte or integer array as alarms have the value 0,1 or 2 (acked) and it must be at least 239 (220+19) elements in size. The array ANA_OUT is probably a FLOAT and should have at least 1 element in size.
- Data_Array_Offset – The first element from table 11 will be placed in element 220 (the 221st element) of the array ALARMS.
- Function – Read polls are generated on a continuous basis.
- Address – The offset into the polled table. Refer to Canatal International Corporation Inc. Product Literature determines the required offset.
- Length – The number of items being read. Not the number of bytes being read.
- Cana_Series & Cana_Table – Enter the series of the device to poll and table number that contains the data of interest. Refer to **Section 9.1**.
- Scan_Interval – Polls will occur every second.

6 Server Side Configuration

For detailed information on FieldServer configuration, refer to the instruction manual for the FieldServer and 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” files on the driver diskette and “.csv” sample files provided with the FieldServer).

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

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for (Driver Name) 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 as default.

6.1 Server 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 ¹ |
| Baud* | Specify baud rate. | 1200 , 2400, 4800, 9600 (Vendor limitation) |
| Parity* | Specify parity. | Even |
| Data_Bits* | Specify data bits. | 7 |
| Stop_Bits* | Specify stop bits. | 1 |
| Protocol | Specify protocol used. | Canatal, Satchnet |

Example

```
// Server Side Connections
Connections
Port          , Baud   , Parity   , Protocol  , Data_Bits  , Stop_Bits
R1           , 1200  , Even    , Canatal   , 7          , 1
```

6.2 Server Side Node Parameters

| 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 (Canatal device address). | 1-63 |
| Protocol | Specify protocol used. | Canatal, Satchnet |

Example

```
// Server Side Nodes
Nodes
Node_Name      , Node_ID  , Protocol
Engine1       , 33      , CATM5X
```

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

6.3 Server Side Map Descriptor Parameters

6.3.1 FieldServer Specific Map Descriptor Parameters

| Column Title | Function | Legal Values |
|---------------------|---|---|
| Map_Descriptor_Name | Name of this Map Descriptor. | Up to 32 alphanumeric characters ² |
| Data_Array_Name | Name of Data Array where data is to be stored in the FieldServer. | One of the Data Array names from Section 4 |
| Data_Array_Offset | Starting location in Data Array. | 0 to (Data_Array_Length-1) as specified in Section 4 |
| Function | Function of Server Map Descriptor. | Passive |

6.3.2 Driver Specific Map Descriptor Parameters

| Column Title | Function | Legal Values |
|--------------|--|--|
| Node_Name | Name of Node to fetch data from. | One of the Node names specified in Section 6.2 |
| Length | Length of Map Descriptor. | 1 - 1000 (Maximum value is dependent on the values of Cana_Series and Cana_Table); Refer to Section 9.1 |
| Address | This term is used to refer to the offset into the Canatal device data table. Each table first element is addressed as zero, the 2nd element as 1 etc. | 0,1,2,3 ... The address & length must not exceed the maximum length of the table being addressed |
| Cana_Series | Refers to the Canatal Corporation Inc. Device Series Number. | 2,4,5,6,9 |
| Cana_Table | The data table in the device to be read. | Non zero integers; Refer to Section 9.1 |
| Cana_Format* | Used to over-ride the default data storage method the driver assigns to the table. This parameter may be required when reading data from a table with mixed data formats. See Section 9.1 to identify which tables have mixed data formats and what the default type the driver allocates to the table. | Byte, BCD, BCD4, Float, Word, Percent, Swap_int, - |

6.3.1 Timing Parameters

| Column Title | Function | Legal Values |
|--------------------|---|--------------|
| Scada_Hold_Timeout | Specifies time Server side waits before responding to Client that Node is offline on FieldServer Client side. | >1.0s |

² Object_Name values of any length may be written via BACnet (subject only to memory and message length constraints).

6.4 Map Descriptor Example

In this example the FieldServer is emulating a Canatal device addressed as node 11. It is not possible to have the FieldServer emulate two devices of different series addressed as node 11.

| Map_Descriptor_Name | Data_Array_Name | Data_Array_Offset | Function | Node_Name | Address |
|---------------------|-----------------|-------------------|----------|-----------|---------|
| DEV11_TBL5 | DA_AI3 | 0 | Passive | Node_12 | 0 |

| Length | Cana_Series | Cana_Table |
|--------|-------------|------------|
| 3 | 4 | 5 |

In the above example:

- Function – Server side Map Descriptors are passive. They respond to read / write commands from a Canatal master.
- Node_Name – The node name connects this Map Descriptor to a node specification.
- Address – Corresponds to the offset into the data table being requested. If the request is for offset 2 (3rd element of the table), then the FieldServer responds with data from array DA_AI3 element 10 (offset into array)+3 (3rd element). If the request was for offset 5 then this Map Descriptor would not be capable of processing the response (starts at 0 and is 3 elements long).
- Cana_Series – The Canatal device series number the FieldServer will emulate.
- Cana_Table – The number of the Canatal device's data table the FieldServer will emulate.

7 Useful Features

7.1 Retrieving Data from Tables with Mixed Data Formats

The driver uses the Data Format listed in **Section 9.1** as the default. If this format is used to store all the data, unexpected results will occur. Read the Canatal product literature, to determine the table storage format of the point(s) of interest and over-ride the default format using the Cana_Format Parameter.

A separate Map Descriptor is required to poll data of each Data Type where mixed formats are stored in a table.

7.1.1 Map Descriptor Example

In this example two Map Descriptors are provided. They both read table 5 of a series 9 device. The default format used by the Driver is Byte, but the data is stored in mixed formats. Thus it is necessary to consult the Canatal literature to determine the format of the required data and specify the Cana_Format keyword to override the default data format. Refer to **Section 9.1**.

| Map_Descriptor_Name | Data_Array_Name | Data_Array_Offset | Function | Node_Name | Address | Length |
|---------------------|-----------------|-------------------|----------|-----------|---------|--------|
| TABLE05_1 | TBL05_DATA | , 0 | , Rdbc | , DEV11 | , 1 | , 7 |
| TABLE05_2 | TBL05_DATA | , 0 | , Rdbc | , DEV11 | , 10 | , 1 |

| , Cana_Series | , Cana_Table | , Scan_Interval | , Cana_Format |
|---------------|--------------|-----------------|---------------|
| , 9 | , 5 | , 1.0s | , - |
| , 9 | , 5 | , 1.0s | , Swap_Int |

In the above example:

- Data_Array_Offset – With the offset equal to zero the data is placed in the array element equal to the address. The offset is added to the address when determining the element of the data array to be used for storage.
- Address – This is the offset of the Boiler Limit Time in Table 5.
- Length, Cana_Series & Cana_Table – Reads elements 1 through 7 from table 5 in a series 9 device. According to the product specification the 8th element is stored using a different format and thus using this Map Descriptor to read mixed data type will produce unexpected results.
- Cana_Format – This parameter over-rides the default data type. According to the product specification the **Boiler Limit Time** is stored as a **Swapped Integer**.

8 Troubleshooting

8.1 Improving Communication on Legacy Canatal Devices

For Canatal Device Series 2-5, setting the Line_Drive_Off connection parameter to 0 helps to reduce the number of errors on connection and could improve communication.

8.2 Driver Error Messages

The following messages are produced by the driver. Those marked *FYI* are for the user's information only and do not require any corrective action. Those marked *Error* require some action from the user.

| Error # | Message | Description |
|---|--|--|
| CANA:#1 | FYI. Station illegal -Forcing from %d to 59 | Valid station numbers are 1-63. |
| CANA:#2 | Error. Series & table numbers are rqd. Correct MapDesc=<%s> | The Map Descriptor parameters Cana_Series and Cana_Table need to be specified. |
| CANA:#3 | Error. Illegal Series Number. Correct MapDesc=<%s> | The series number specified cannot be processed by this driver. Read the vendor documentation and correct the CSV file. |
| CANA:#4 | Error. Illegal Table Number(%d). Correct MapDesc=<%s> | All Canatal equipment series do not have the same number of tables. Read the vendor documentation and correct the CSV file. |
| CANA: #5 | Error. Series & table numbers are rqd. Empty Table. Correct MapDesc=<%s> | The series & table combinations specified are not valid. Read the vendor documentation and correct the CSV file. |
| CANA:#6 | Error. Start address is beyond end of table. MapDesc=<%s> | The address is the equivalent term to offset in the Vendor literature. Different tables have different lengths. The lengths may also be different based on the series number of the equipment. Read the vendor documentation and correct the CSV file. |
| CANA:#7 | FYI. Requested too many elements from table. MapDesc=<%s> Changed request from %u to %u | The driver has truncated the Map Descriptor length so that it does not read/write beyond the end of a table. |
| Errors 8 - 12 are produced when parsing a response from a Canatal device. They may result from a corrupted message. If they occur infrequently then they require no attention. Check the communication stats. | | |
| CANA:#8 | Error. Poll(%d)/Response(%d) Stns Not Equal | The station polled has responded but as a different station. The response will be abandoned. This may mean that the device's address has been incorrectly set or that the device is not functioning correctly. |
| CANA:#9 | Error. Poll(%d)/Response(%d) Tables Not Equal | The table polled has responded but as a different table. The response will be abandoned. This may mean that the device's address has been incorrectly set or that the device is not functioning correctly. |

Additional Information

| Error # | Message | Description |
|----------|--|---|
| CANA:#10 | Error. Response Table=%d Unknown | The table number specified cannot be processed by this driver. Read the vendor documentation and correct the CSV file. |
| CANA:#11 | Error. Poll(%d)/Response(%d) Address Not Equal | The address polled has responded but as a different address. The response will be abandoned. This may mean that the device's address has been incorrectly set or that the device is not functioning correctly. |
| CANA:#12 | Error. Response Expected(%d)/Actual(%d) Bytes | The data was requested from a table with mixed data formats and the data served was not in Byte (default) format. Use the Cana_Format parameter to override the Data Format in the CSV file. |
| CANA:#13 | Error. Array is too small for mapDesc=<%s> Some data will not be stored. | This is regarded as an error as the Map Descriptor has requested more data than can be stored. The excess data has been discarded. Modify the Map Descriptor definition or change the array length in the CSV file and restart the FieldServer. |
| CANA:#14 | Error. Unknown Data method (%d) for mapDesc=<%s> | Report this error to FieldServer support. |
| CANA:#15 | Error. Unknown Data method (%d) for mapDesc=<%s> | |
| CANA:#16 | Error. Ack Expected | |

9 Reference

9.1 Canatal Device Data Tables

Canatal Devices have their data organized into data tables. A query requests a number of elements from a particular data table. The number, length and content of data tables vary according to the series number of the device being addressed. The following series and tables are supported by this driver.

NOTE: For Canatal firmware revisions from 2002 onwards, “Time” data available in Table 13 is now available in Table 69, and Table 13 has been made obsolete.

| Table | Data Format | Max Elements | Read/Write | Description | Notes |
|-----------------|----------------|--------------|------------|-----------------------|-------|
| Series 2 | | | | | |
| 4 | CANA_BCD_CODED | 6 | R | FirmWare | |
| 5 | CANA_FLOAT | 34 | RW | Keycode | |
| 6 | CANA_FLOAT | 6 | R | Sensor Reading | |
| 7 | CANA_FLOAT | 3 | RW | Security | |
| 8 | CANA_BYTE | 84 | RW | Time Schedule | |
| 9 | CANA_BYTE | 2 | R | Switched Output | |
| 10 | CANA_PERCENT | 6 | R | Analog Outputs | |
| 11 | CANA_BYTE | 19 | RW | Alarm Status | |
| 12 | CANA_BYTE | 1 | R | Control Status | |
| 13 | CANA_BCD_CODED | 7 | RW | Time | |
| 14 | CANA_BYTE | 4 | R | Staged Output Status | |
| 15 | CANA_BYTE | 1 | RW | Plant Status | |
| Series 4 | | | | | |
| 4 | CANA_BCD_CODED | 6 | R | FirmWare | |
| 5 | CANA_FLOAT | 34 | RW | Keycode | |
| 6 | CANA_FLOAT | 6 | R | Sensor Reading | |
| 7 | CANA_FLOAT | 3 | RW | Security | |
| 8 | CANA_BYTE | 1 | R | Switched Output | |
| 9 | CANA_PERCENT | 5 | R | Analog Outputs | |
| 10 | CANA_BYTE | 16 | RW | Alarm Status | |
| 11 | CANA_BYTE | 1 | R | Control Status | |
| 12 | CANA_BYTE | 4 | R | Staged Output Status | |
| 13 | CANA_BYTE | 1 | RW | Plant Status | |
| Series 5 | | | | | |
| 4 | CANA_BCD_CODED | 6 | R | FirmWare | |
| 5 | CANA_SWAP_INT | 100 | RW | Keycode | |
| 6 | CANA_BYTE | 4 | R | Sensor Reading | |
| 7 | CANA_SWAP_INT | 3 | RW | Security | |
| 8 | CANA_BYTE | 84 | RW | Time Schedule | |
| 9 | CANA_BYTE | 3 | R | Switched Output | |
| 10 | CANA_PERCENT | 4 | R | Analog Outputs | |
| 11 | CANA_BYTE | 48 | RW | Alarm Status | |
| 12 | CANA_BYTE | 3 | R | Control Status | |
| 13 | CANA_BCD_CODED | 7 | RW | Time | |
| 14 | CANA_BYTE | 4 | R | Staged Output Status | |
| 15 | CANA_WORD | 17 | RW | Accumulating Run Time | |

| Table | Data Format | Max Elements | Read/Write | Description | Notes |
|--------------------------------|----------------|--------------|------------|------------------------|-------|
| 15 | CANA_BYTE | 1 | RW | Machine Remote Control | |
| Series 6 & Series 8 | | | | | |
| 4 | CANA_BCD_CODED | 6 | R | FirmWare | |
| 5 | CANA_BYTE | 75 | RW | Keycode | Mixed |
| 6 | CANA_SWAP_INT | 7 | R | Sensor Reading | |
| 7 | CANA_BCD_4 | 3 | RW | Security | |
| 8 | CANA_BYTE | 84 | RW | Time Schedule | |
| 9 | CANA_BYTE | 16 | R | Switched Output | |
| 10 | CANA_PERCENT | 4 | R | Analog Outputs | |
| 11 | CANA_BYTE | 241 | RW | Alarm Status | |
| 12 | CANA_BYTE | 10 | R | Control Status | |
| 13 | CANA_BCD_CODED | 7 | RW | Time | |
| 14 | CANA_BYTE | 4 | R | Staged Output Status | |
| 15 | CANA_SWAP_INT | 64 | RW | Accumulating Run Time | |
| 16 | CANA_BYTE | 1 | RW | Machine Remote Control | |
| 17 | CANA_BYTE | 7 | RW | KeyCode2 | Mixed |
| Series 9 | | | | | |
| 4 | CANA_BCD_CODED | 6 | R | FirmWare | |
| 5 | CANA_BYTE | 75 | RW | Keycode | Mixed |
| 6 | CANA_SWAP_INT | 7 | R | Sensor Reading | |
| 7 | CANA_BCD_4 | 3 | RW | Security | |
| 8 | CANA_BYTE | 84 | RW | Time Schedule | |
| 9 | CANA_BYTE | 16 | R | Switched Output | |
| 10 | CANA_PERCENT | 4 | R | Analog Outputs | |
| 11 | CANA_BYTE | 121 | RW | Alarm Status | |
| 12 | CANA_BYTE | 10 | R | Control Status | |
| 13 | CANA_BCD_CODED | 7 | RW | Time | |
| 14 | CANA_BYTE | 4 | R | Staged Output Status | |
| 15 | CANA_SWAP_INT | 64 | RW | Accumulating Run Time | |
| 16 | CANA_BYTE | 1 | RW | Machine Remote Control | |
| 17 | CANA_BYTE | 7 | RW | KeyCode2 | Mixed |

9.2 Recommended Data Array Formats Based on Canatal Data Types

| Data Type | Suitable Data Array Format |
|----------------|---------------------------------------|
| CANA_BCD_CODED | Byte, Uint16, Uint32, Float |
| CANA_BYTE | Byte, Uint16, Uint32, Float |
| CANA_SWAP_INT | Sint16, Sint32, Uint16, Uint32, Float |
| CANA_BCD_4 | Byte |
| CANA_WORD | Uint16, Uint32, Float |
| CANA_FLOAT | Float |