Driver Manual

FS-8700-26 Notifier AFP 200/300/400

APPLICABILITY & EFFECTIVITY
Effective for all systems manufactured after June 2023.

Driver Revision: 1.02
Document Revision: 7.B

MSA Safety
1000 Cranberry Woods Drive
Cranberry Township, PA 16066 USA
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

The Notifier AFP 200/300/400 driver allows the FieldServer to transfer data to and from the Notifier AFP 200, AFP 200, and AFP 400 Automatic Fire Alarm Panels over RS-232 using Notifier AFT 200/300/400 protocol. The panel MUST output messages in English. The FieldServer functions as a Client with this driver.

The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer. As the AFP 200 panel interface differs in many aspects from the AFP 300/ AFP 400 interface, these panel interfaces will be discussed separately.

2 Driver Scope of Supply

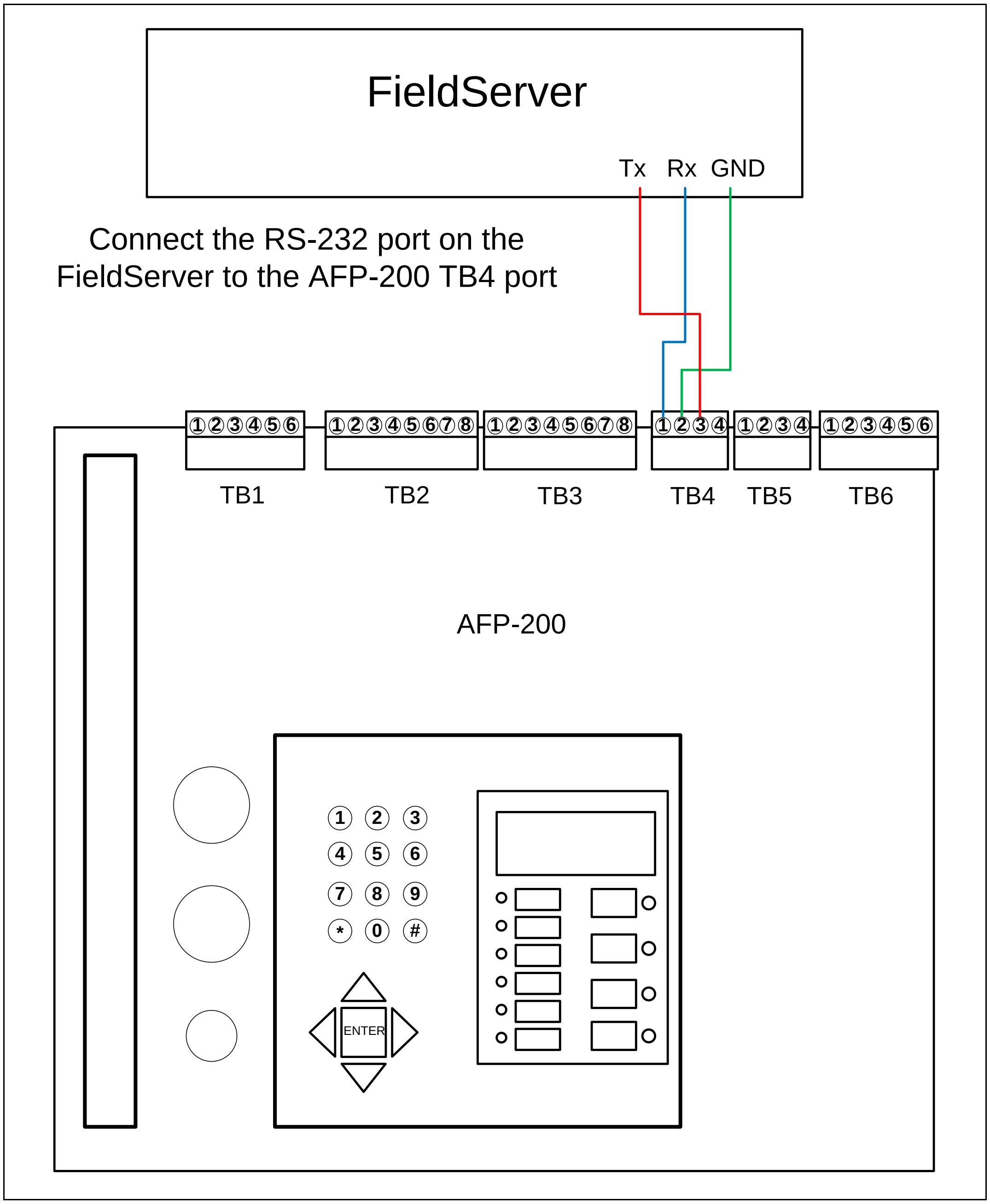
2.1 Provided by the Supplier of 3rd Party Equipment

2.1.1 Required 3rd Party Hardware

Description
Notifier AFP 200, AFP 300, AFP 300 Fire Panel

3 Hardware Connections AFP 200

It is possible to connect a Notifier AFP 200 device to any serial port. These ports just need to be configured for Notifier AFP 200 in the configuration file.



Connect the RS-232 port on the FieldServer to the AFP-200 TB4 port

Connecting Pinouts

FieldServer	TB4 (AFP-200)	
Tx	3	Rx
Rx	1	Tx
GND	2	REF

4 Data Array Parameters AFP 200

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_Format	Provides data format.	Float, Bit, Byte, Uint16, Uint32, Sint16, Sint32
Data_Array_Length	Number of Data Objects.	1-10000

Example

```
//    Data Arrays
Data_Arrays

Data_Array_Name    , Data_Format    , Data_Array_Length
Det_01             , Bit            , 500
Mod_01             , Bit            , 500
Det_02             , Bit            , 500
Mod_02             , Bit            , 500
DA_Zones           , Bit            , 100
DA_System          , Bit            , 16
```

5 Client Side Configuration for AFP 200

The Notifier driver is both active and passive in that it will continuously scan for status' requested in the Client side Map Descriptor section of the configuration file (active element). In addition, it will receive all unsolicited messages sent by the panel to the FieldServer (passive element).

Passive Elements (unsolicited panel messages) usually constitute alarms and other time sensitive signals. These signals are acquired by the FieldServer on an interrupt basis and as such, are available immediately in the FieldServer Data Arrays.

Active elements (polled messages) usually constitute panel and point status updates. Due to the fixed baud rate of communication channel, polled points are updated relatively slowly (about 2 seconds per point). Polled messages are considered to be a background activity, and generally this function is used to ensure that the database of the FieldServer (Data Arrays) remains consistent with the Notifier Database.

All Data Array elements are reset when the system reset button on the Notifier panel is pressed. In addition, certain specific Data Array Elements will be reset when their related "Normal/Inactive" message is received.

NOTE: In the tables below, * indicates an optional parameter, with the bold legal value as default.

5.1 Client Side Connection Parameters

In this section of the configuration, the port (and its associated properties) being used for connection to the Notifier panel is defined.

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.	2400
Parity*	Specify parity.	Even
Data_Bits*	Specify data bits.	7
Stop_Bits	Specify number of stop bits.	1
Protocol	Specify protocol.	Notifier – AFP
Timeout	Specify maximum response time.	>10 S
IC_Timeout	Specify inter character timeout.	>1.0 S

Example

```
//   Client Side Connections

Connections
Port   , Baud    , Parity   , Data_Bits    , Stop_Bits    , Protocol            , Timeout     , IC_Timeout
P1     , 2400    , Even     , 7            , 1            , NOTIFIER-AFP        , 10.0s       , 1.0s
```

[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.

5.2 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	Identification number for this Node.	The Node_ID gets set to the Node_ID configured in the AFP panel
Protocol	Specify Protocol used.	Notifier-AFP
Port	Specify through which port the device is connected to the FieldServer.	P1-P2, R1-R2[2]
Timeout	Specify maximum response time.	>10S

Example

```
Nodes
Node_Name    , Node_ID    , Protocol        , Port    , Timeout
AFP_1        , 0          , NOTIFIER-AFP    , P1      , 10.0s
```

5.3 Client Side Map Descriptor Parameters

Map Descriptors determine where in the Data Arrays the various points from the Notifier panel will be mapped (stored).

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 “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.	RDBC, Passive

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 “Client Node Descriptor” above
Device_Type	Device Type in Panel.	Detector, Module, Zone, Bell, System
Data_Type	Data type in Panel.	Refer to **Section 9.1**
Address	Starting address of read block.	Refer to **Section 9.2**
Length	Number of modules, alarms, etc.	1-99

5.3.3 Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Specify minimum Interval between Scans.	> 10.0 S

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

5.4 Map Descriptor Example

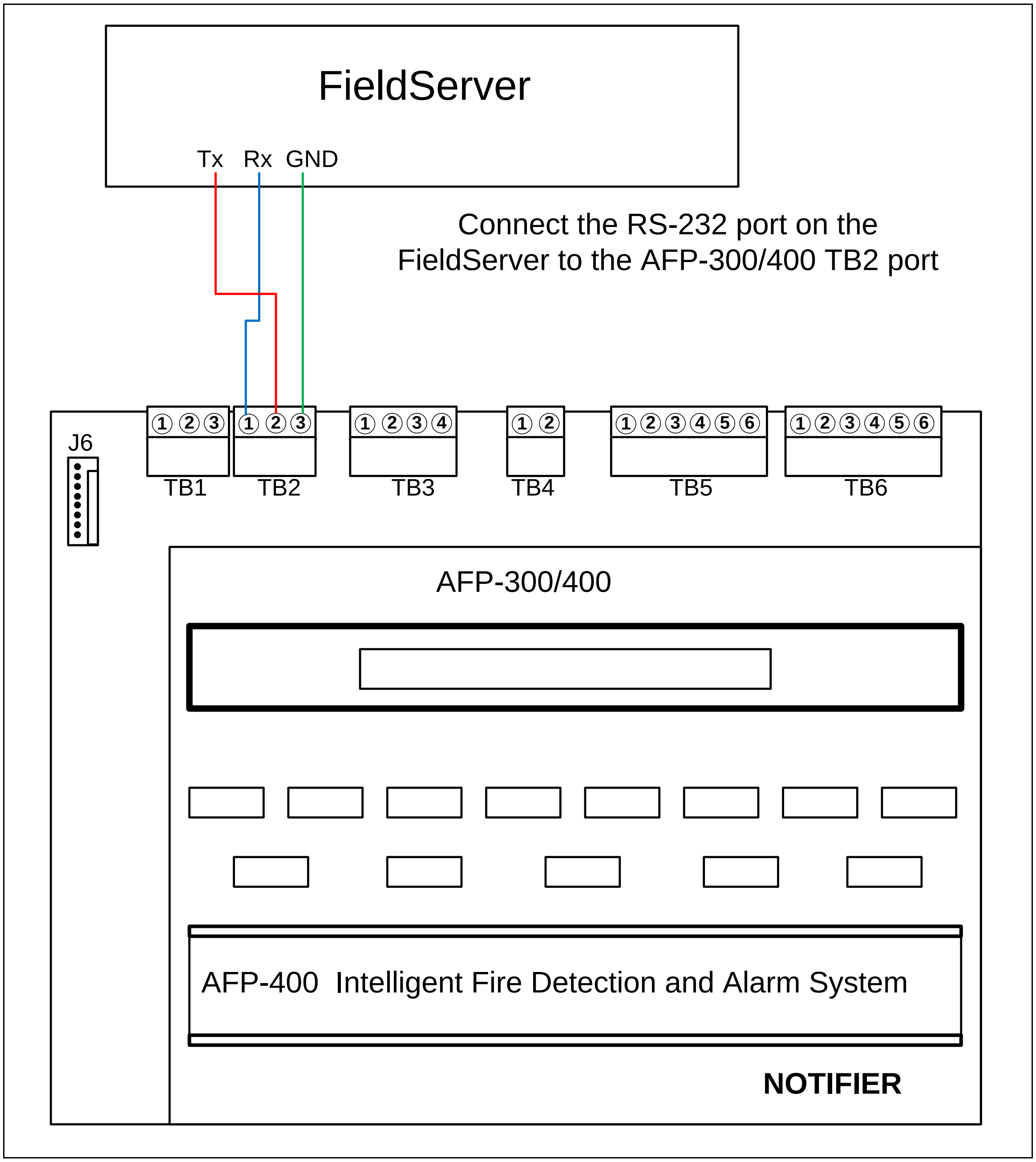
```
//    Client Side Map Descriptors

Map Descriptors
Map_Descriptor_Name    , Data_Array_Name    , Data_Array_Offset    , Function    , Device_Type    , Data_Type
AFP200_det_alm_01      , Det_01             , 1                    , Rdbc        , Detector       , Alarm
AFP200_det_trb_01      , Det_01             , 101                  , Passive     , Detector       , Trouble
AFP200_det_pas_01      , Det_01             , 201                  , Passive     , Detector       , Pre_Alarm
AFP200_det_nin_01      , Det_01             , 301                  , Passive     , Detector       , Not_Installed
AFP200_mod_alm_01      , Mod_01             , 1                    , Rdbc        , Module         , Alarm
AFP200_mod_trb_01      , Mod_01             , 101                  , Passive     , Module         , Trouble
AFP200_mod_pas_01      , Mod_01             , 201                  , Passive     , Module         , Pre_Alarm
AFP200_mod_nin_01      , Mod_01             , 301                  , Passive     , Module         , Not_Installed
AFP200_mod_nal_01      , Mod_01             , 401                  , Passive     , Module         , Non_Alarm
```

```
, Node_Name    , Address    , Length    , Scan_Interval
, AFP_1        , 01         , 99        , 2
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99        , 2
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
```

6 Hardware Connections AFP 300/400

It is possible to connect a Notifier AFP 300 or AFP 400 device to any serial port. These ports just need to be configured for Notifier AFP 300/400 in the configuration file.



Connect the RS-232 port on the FieldServer to the AFP-300/400 TB2 port

Connecting Pinouts

FieldServer	TB2 (AFP-300/400)	
Tx	2	Tx
Rx	1	Rx
GND	3	GND

7 Data Array Parameters AFP 300/400

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_Format	Provides data format.	Float, Bit, Byte, Uint16, Uint32, Sint16, Sint32
Data_Array_Length	Number of Data Objects.	1-10000

<u>Example</u>

```
//    Data Arrays
Data_Arrays
Data_Array_Name       , Data_Format      , Data_Array_Length
Det_01                , Bit              , 500
Mod_01                , Bit              , 500
Det_02                , Bit              , 500
Mod_02                , Bit              , 500
DA_Zones              , Bit              , 100
DA_System             , Bit              , 1
```

8 Client Side Configuration for AFP 300/400

The Notifier driver is both active and passive in that it will continuously scan for status' requested in the Client side Map Descriptor section of the configuration file (active element). In addition, it will receive all unsolicited messages sent by the panel to the FieldServer (passive element).

Passive Elements (unsolicited panel messages) usually constitute alarms and other time sensitive signals. These signals are acquired by the FieldServer on an interrupt basis and as such, are available immediately in the FieldServer Data Arrays.

Active elements (polled messages) usually constitute panel and point status updates. Due to the fixed baud rate of the communication channel, polled points are updated relatively slowly (about 8 seconds per point). Therefore, if the status of a large number of points is desired, it could mean that it could take minutes for the status of a point to update in the Data Array, e.g. an alarm point that appeared immediately in the data array due to an unsolicited message from the panel would only clear when the point's status is updated and the alarm has cleared in the panel. As the alarm was reported immediately, this does not compromise the integrity of the system.

NOTE: In the tables below, * indicates an optional parameter, with the bold legal value as default.

8.1 Client Side Connection Parameters

In this section of the configuration, the port (and its associated properties) being used for connection to the Notifier panel is defined.

Section Title
Connections

Column Title	Function	Legal Values
Port	Specify which port the device is connected to the FieldServer.	P1-P2, R1-R2[3]
Baud*	Specify baud rate.	2400
Parity*	Specify parity.	None
Data_Bits*	Specify data bits.	8
Stop_Bits	Specify number of stop bits.	1
Protocol	Specify protocol.	Notifier – AFP
Timeout	Specify maximum response time.	>10 S
IC_Timeout	Specify inter character timeout.	>1.0 S

Example

```
//   Client Side Connections

Connections
Port   , Baud   , Parity   , Data_Bits   , Stop_Bits   , Protocol       , Timeout   , IC_Timeout
P1     , 2400   , None     , 8           , 1           , NOTIFIER-AFP   , 10.0s     , 1.0s
```

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

8.1 Client Side Node Descriptors

The FieldServer automatically assigns the Device Internal Node_ID of station 257.

Section Title		
Nodes		
Column Title	**Function**	**Legal Values**
Node_Name	Provide name for node.	Up to 32 alphanumeric characters
Node_ID	Identification number for this Node.	The Node_ID gets set to the Node_ID configured in the AFP panel
Protocol	Specify Protocol used.	Notifier-AFP
Port	Specify through which port the device is connected to the FieldServer.	P1-P2, R1-R2[4]
Timeout	Specify maximum response time.	>10S

Example

```
Nodes
Node_Name    , Node_ID    , Protocol          , Port    , Timeout
AFP_1        , 0          , NOTIFIER-AFP      , P1      , 10.0s
```

8.2 Client Side Map Descriptor Parameters

Map Descriptors determine where in the Data Arrays the various points from the Notifier panel will be mapped (stored).

8.2.1 FieldServer Related Map Descriptor Parameters

Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor.	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer.	One of the Data Array names from “Data Array” section above
Data_Array_Offset	Starting location in Data Array.	0 to maximum specified in “Data Array” section above
Function	Function of Client Map Descriptor.	RDBC, Passive

8.2.2 Driver Related Map Descriptor Parameters

Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from.	One of the node names specified in “Client Node Descriptor” above
Device_Type	Device Type in Panel.	Detector, Module, Zone, Bell, System
Data_Type	Data type in Panel.	Refer to **Section 9.1**
Address	Starting address of read block.	Refer to **Section 9.2**
Length	Number of modules, alarms, etc.	1-99

8.2.3 Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Specify minimum Interval between Scans.	> 10.0 S

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

8.3 Map Descriptor Example

8.3.1 General

```
//    Client Side Map Descriptors

Map Descriptors
Map_Descriptor_Name    , Data_Array_Name    , Data_Array_Offset    , Function    , Device_Type    , Data_Type
AFP200_det_alm_01      , Det_01             , 1                    , Rdbc        , Detector       , Alarm
AFP200_det_trb_01      , Det_01             , 101                  , Passive     , Detector       , Trouble
AFP200_det_pas_01      , Det_01             , 201                  , Passive     , Detector       , Pre_Alarm
AFP200_det_nin_01      , Det_01             , 301                  , Passive     , Detector       , Not_Installed
AFP200_mod_alm_01      , Mod_01             , 1                    , Rdbc        , Module         , Alarm
AFP200_mod_trb_01      , Mod_01             , 101                  , Passive     , Module         , Trouble
AFP200_mod_pas_01      , Mod_01             , 201                  , Passive     , Module         , Pre_Alarm
AFP200_mod_nin_01      , Mod_01             , 301                  , Passive     , Module         , Not_Installed
AFP200_mod_nal_01      , Mod_01             , 401                  , Passive     , Module         , Non_Alarm
```

```
, Node_Name    , Address    , Length    , Scan_Interval
, AFP_1        , 01         , 99        , 2
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99        , 2
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
, AFP_1        , 01         , 99
```

8.3.2 Store Trouble Status of System

This Map Descriptor will store the Trouble Status of the entire system. The driver looks for "TROUBL IN SYSTEM" in a message from the panel and stores a 1 at the determined offset to indicate that there is a trouble in the system. Since this is a discrete point, a Data Array length of 1 is sufficient. The Data Array will be reset on "SYSTEM RESET" or "ALL SYSTEMS NORMAL".

```
//    Client Side Map Descriptors

Map Descriptors
Map_Descriptor_Name    , Data_Array_Name    , Data_Array_Offset    , Function    , Device_Type    , Data_Type
AFP200_det_trb_01      , DA_System          , 101                  , Passive     , system         , Trouble
```

```
, Node_Name    , Address    , Length    , Scan_Interval
, AFP_1        , -          , 99
```

9 Special Driver Parameters

9.1 Notifier Data Types

It is possible to scan separately for the various types of data points in the Notifier panel. In particular, modules have several data types known as “type codes.” (Refer to the relevant operations manual, for a detailed description of the behavior of each of the type codes).

Permissible parameters for the Data_Type field in the Notifier Map Descriptor are as follows:

Alarm
Trouble
Non_Alarm
Tamper
Hazard Alert
Abort_Switch
Evacuate
Second_Shot

Pre_Alarm
Not_Installed
Supervisory
Non_Fire
Fire_Control
Silence
PAS_Inhibit
Security

These data types relate to the type codes as follows.

Alarm Data Type
This data type is valid for the following Device Types:

- Detector
- Module
- Zone
- Bell

A Map Descriptor with this data type declared will set a bit in the designated data array at the designated offset if any one of the following type codes are assigned to the related Notifier SLC address in the Notifier Panel and an alarm is active at that point. Alarms are reset by pressing System Reset on the Notifier Panel.

Monitor
Smoke Detect
Blank
Man. Release
Tele Page
Man. Rel. Delay
Comb. Monitor

Pull Station
Heat Detect
Waterflow
Trouble Man.
All call page
Sprinkler Sys
All Detector Types

Trouble Data Type
This data type is valid for the following device types:

- Detector
- Module
- System

A Map Descriptor with this data type declared will set a bit in the designated data array at the designated offset if the related SLC network device goes into trouble status. Troubles are reset by pressing System Reset on the Notifier Panel. See example in **Section** 8.3.2.

Pre_Alarm Data Type

This data type is valid for the following device types:

- Detector
- Module

A Map Descriptor with this data type declared will set a bit in the designated data array at the designated offset if the pre-alarm for a device becomes active. Pre_Alarms are reset by pressing System Reset on the Notifier Panel.

Not_Installed Data Type

This Data Type is valid for the following device types:

- Detector
- Module

A Map Descriptor with this data type declared will set a bit in the designated data array at the designated offset if there is no device installed at the related SLC network address. The “Not_Installed” status of point is continuously updated by the panel.

Non_Alarm Data Type

This data type is valid for the following device types:

- Modules

A Map Descriptor with this data type declared will set a bit in the designated data array at the designated offset if any one of the following type codes are assigned to the related SLC address in the Notifier Panel, and the status of that address is active:

Supervisory
Non_Fire
Fire_Control
Silence
Evacuate
Second_Shot
Tamper
Hazard_Alert
Abort_Switch
System_Reset
PAS_Inhibit
Security

It is also possible to use each of these type codes as a data type, in which case only the related type code will activate a bit in the data array. Depending on the nature of the non_alarm type code, active status’ are reset either by inactive status reported, or by System Reset being pressed, or both.

9.2 Permissible Addresses

Permissible addresses for the various device types are as follows:

AFP 200		AFP 300		AFP 400	
Detectors	1-99	Detectors	101-199	Loop 1 Detectors	101-199
Modules	1-99	Modules	101-199	Loop 2 Detectors	201-299
Zones	1-99	Zones	101-199	Loop 1 Modules	101-199
Bells	1-4	Bells	1-4	Loop 2 Modules	201-299
System	--	System	--	Zones	1-99
				Bells	1-4
				System	--

9.3 Zones

If it is necessary for the related zone to be scanned by the FieldServer, then the Zone must have its default text programmed in the Notifier Panel, i.e. “Zone xx”, where xx is the zone number. A Map Descriptor must also exist to map the required zones to a designated data Array.

Active zones are reset only when the system reset button is pressed on the Notifier panel.

NOTE: The Notifier protocol only reports the first of the five possible zones for any point in the FieldServer. It is advisable to assign the five zones for every point with this in mind.

10 Driver Notes

10.1 Using Log Files to Test the Driver

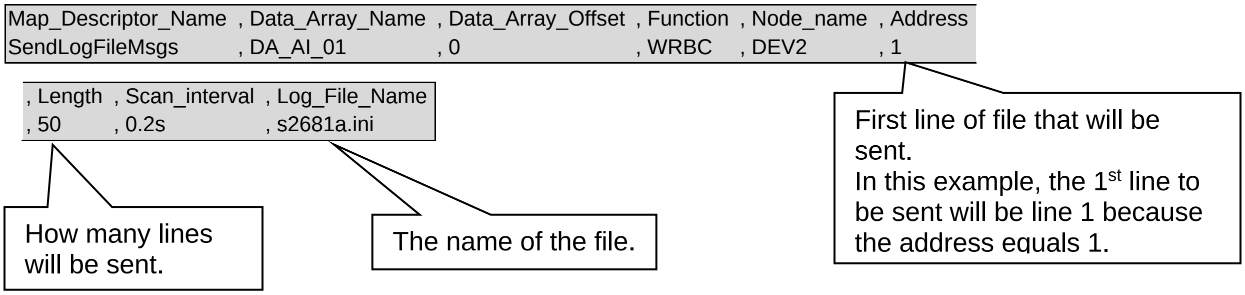
These notes are intended for FieldServer tech support only.

It is possible to construct a test script using an ASCII file to send messages to the driver. This feature was added in version 1.02a of the driver.

This is an example of a log file. You can see the file is an ASCII file. Lines that begin ‘***//***’ are ignored.

```
// --------------------------------
// Comments begin with a double slash
//
// Specify the file name in the Client map descriptor using the "Log_File_Name" parameter
// Set the address equal to the 1st line of the file to be sent
// Set the length equal to the number of lines to be sent
// when the final line is sent then no more messages are sent
// The driver removes up to 2 cr's and 2 lf's chars looking backward from the end of the line
//   and then appends one crlf pair.
// Except when the 1st byte of the line's ascii value is less than 32
~A
1
D005
```

The following example illustrates the usage.



```
Map_Descriptor_Name  , Data_Array_Name  , Data_Array_Offset  , Function  , Node_name  , Address
SendLogFileMsgs      , DA_AI_01         , 0                  , WRBC      , DEV2       , 1

, Length  , Scan_interval  , Log_File_Name
, 50      , 0.2s           , s2681a.ini
```

How many lines will be sent.

The name of the file.

First line of file that will be sent.
In this example, the 1st line to be sent will be line 1 because the address equals 1.

11 Troubleshooting

If the FieldServer reboots when connected to the Panel Serial port, then it is most likely that an Optical Isolator is required to balance ground potential differences. Such differences have been known to damage the FieldServer serial port, and therefore it is recommended that this action is taken as soon as the symptom is observed.

12 Driver Error Messages

When a message is marked with a '*' this means that when the same error occurs again the message will be suppressed. This is done to stop the error log being filled with duplicate messages which do not convey additional information.

Message	Explanation
NAFP:#01 Err. Need an MD to store detector msg. Detector=102 Device_Type=101 Data_Type=Not_Installed	The message is printed when the driver is trying to store information about a detector and cannot find a Map Descriptor to use for the storage. The message reports the detector number, the device type (see message #02 for device types) and the Data_Type. This information is sufficient to create a new MapDesc to define a storage location. In the example the Client polled for detector data - the panel responded that the detector wasn't installed and no MD was defined to store 'Not_Installed' responses. Each time one of these errors occur the driver increments the MSG_IGNORED stat. This stat does not increment the error count on the connection overview page of FS-GUI.[5]
NAFP:#02 FYI. Device_Types: Det=101 Mod=102 Zone=103 Bell=104 Sys=105	This message is for your information only and does not require corrective action on your part.
NAFP:#03* Err. Unknown Type. Expected D/M/Z/B. Rcvd=%c=0x%x	There is no corrective action you can take to eliminate this error. If the error occurs rarely then it may be the result of a corrupted message – this can occur from noise on the communication line. If it occurs repeatedly then you need to take a log and report the problem to tech Support. A procedure for taking a log is provided in the supplied FieldServer manual. The error occurs when parsing a message.
NAFP:#04 FYI. Emulating 2003 Firmware.	This message is for your information only and may be safely ignored. The message reports that the Server side of the driver is emulating the new (summer 2003) panel firmware.
NAFP:#05a Err. MD too short. No response will be sent. Rqd Offset=%d Actual Range=%d to %d	The Server side of the driver produces this message if the address range of a Server MD is too short to serve a response for the device/module etc. requested.5
NAFP:#06 Err. Cant Store. RqdOffset=%d DA=%s Length=%d	The offset into the data array required to store the data is larger than the number of elements in the data array. Adjust the length of the reported Data Array (DA). There are a few variations of this message which are used by tech support to trace the source of the problem. 5

[5] Modify the CSV file, download the modified file to the FieldServer and reset the FieldServer for the changes to take effect.