



A Sierra Monitor Company

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

**FS-8700-141 AOS AINP**

**APPLICABILITY & EFFECTIVITY**

**Effective for all systems manufactured after April 2012**

Driver Version: 1.00  
Document Revision: 3

**TABLE OF CONTENTS**

**1 AOS AINP Description ..... 3**

**2 Driver Scope of Supply ..... 4**

    2.1 Supplied by FieldServer Technologies for this driver .....4

    2.2 Provided by the Supplier of 3<sup>rd</sup> Party Equipment .....4

        2.2.1 *Required 3<sup>rd</sup> Party Hardware* .....4

**3 Hardware Connections..... 4**

**4 Data Array Parameters..... 5**

**5 Configuring the FieldServer as a AOs AINP Client ..... 6**

**6 Configuring the FieldServer as a AOS AINP Server ..... 7**

    6.1 Server Side Connection Parameters .....7

    6.2 Server Side Node Parameters .....8

    6.3 Server Side Map Descriptor Parameters.....8

        6.3.1 *FieldServer Specific Map Descriptor Parameters* .....8

        6.3.2 *Driver Specific Map Descriptor Parameters* .....8

        6.3.3 *Master Parameter Map Descriptor Example.* .....9

        6.3.4 *Master Command Map Descriptor Example.* .....9

        6.3.5 *Slave Parameter Map Descriptor Example.* .....10

        6.3.6 *Slave Command Map Descriptor Example.* .....10

**Appendix A. Useful Features..... 11**

    Appendix A.1. Parameter 0.....11

## 1 AOS AINP DESCRIPTION

The Serial AOS AINP (Advanced Internal Network Proprietary Communication Protocol) driver allows the FieldServer to transfer data to and from devices over RS-485 using AOS AINP protocol. . The *Advanced Internal Network Proprietary Communications Protocol.doc* document forms the basis of the AINP driver.

The FieldServer can be configured as a Server. The Client side has been implemented only for FieldServer's quality assurance requirements.

This driver can be used to transfer parameters and commands to/from AO Smith Master Controllers. Each remote device (e.g. AO Smith Master Controller) can have 30 parameter blocks; each with 255 (16bit integer) parameters which can be stored and updated by the Driver. Each Master device can have a command block of up to 255 commands. The Driver can be configured to execute any of these commands at the Server. The Server can be configured with a combined total of 255 of its own parameters and commands which can be read/written or executed by the Master Device.

### Max Nodes Supported

FieldServer Mode	Nodes	Comments
Client (N/A)	N/A	The Client side has been implemented only for FieldServer's quality assurance requirements.
Server	1	Each FieldServer can act as single AINP Server and can be configured as any (1-31) AINP type.

## 2 DRIVER SCOPE OF SUPPLY

### 2.1 Supplied by FieldServer Technologies for this driver

FieldServer Technologies PART #	Description

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

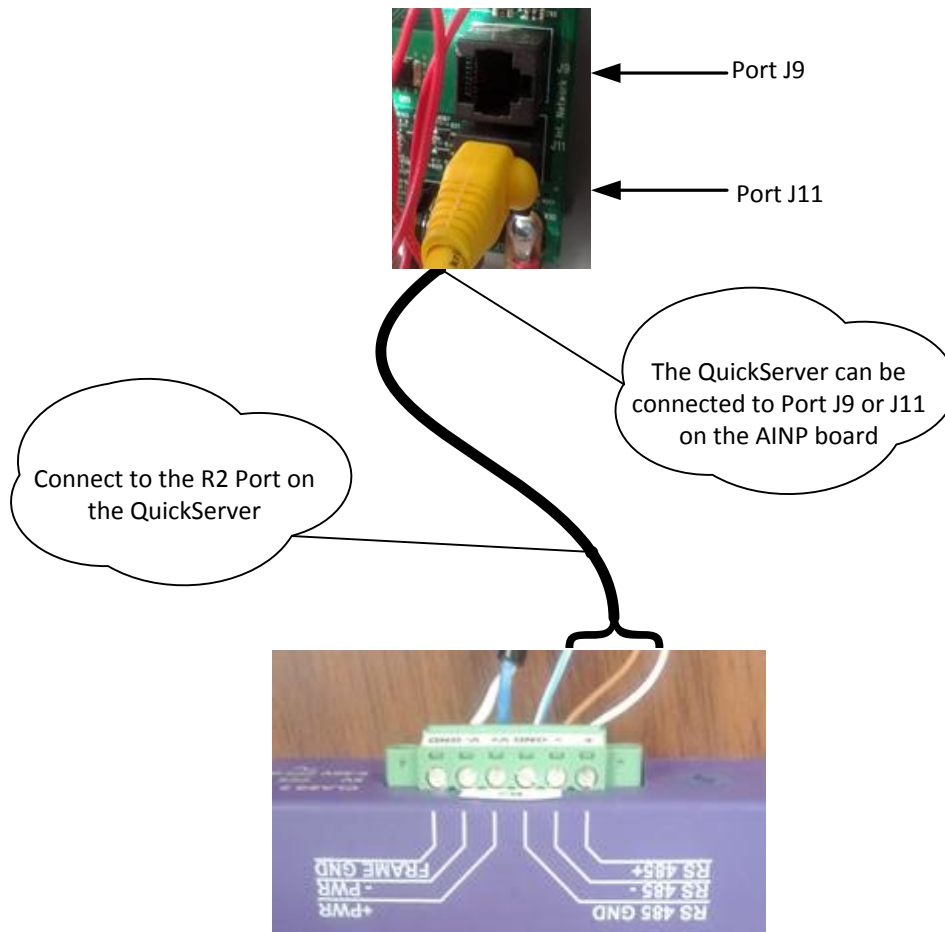
#### 2.2.1 Required 3<sup>rd</sup> Party Hardware

Part #	Description
	AOSmith device supporting AINP Client.
	Connection cable between the FieldServer and AOSmith equipment as shown in the connection diagram.

## 3 HARDWARE CONNECTIONS

The FieldServer is connected to the AOSmith device as shown in the connection drawing.

Configure the AOSmith device according to the manufacturer’s instructions



**Connection Pinouts**

RJ45 Pin#	R2 pin Name	Color
8	RS485-	Brown
7	RS485+	White/Brown
5	GND	Blue/White

**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	Function	Legal Values
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, UInt16, Sint16.
Data_Array_Length	Number of Data Objects. Must be larger than the data storage area required by the Map Descriptors for the data being placed in this array.	1-10,000

**Example**

```
// Data Arrays
Data_Arrays
Data_Array_Name , Data_Array_Format , Data_Array_Length
DA_Mast_Parm , UInt16 , 255
DA_Mast_Cmd , UInt16 , 255
DA_Slv_Parm , UInt16 , 200
DA_Slv_Cmd , UInt16 , 55
```

## 5 CONFIGURING THE FIELDSEVER AS A AOS AINP CLIENT

The Client side of the driver is intended to support FieldServer's Quality Assurance program and is not intended to provide complete emulation of an AOSmith AINP Client and is thus not fully documented. Should you require the Client side functionality to be documented and enhanced, please contact FieldServer's sales group.

## 6 CONFIGURING THE FIELD SERVER AS A AOS AINP SERVER

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 an AOS AINP Client.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for AOS AINP communications, the driver independent FieldServer buffers need to be declared in the “Data Arrays” section, the FieldServer virtual Node(s) needs to be declared in the “Server Side Nodes” section, and the data to be provided to the clients needs to be mapped in the “Server 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.

### 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	R2
Protocol	Specify protocol used	AOS_AINP, AINP
Baud*	Specify baud rate	<b>19200</b> , 38400 (Protocol limitation)
Parity*	Specify parity	<b>None</b> (Protocol limitation)
Data_Bits*	Specify data bits	<b>9</b>
Stop_Bits*	Specify stop bits	<b>1</b> , 2
Server_Hold_Timeout*	Specifies time FieldServer will reserve the Server side connection while waiting for the Client side to update data in the Data_Array	>1.0s, <b>5s</b>

#### Example

```

//      Server Side Connections

Connections
Port   , Protocol   , Baud   , Data_Bits   , Stop_Bits
R2     , AINP           , 19200  , 9            , 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	Specify device type ID	1-31, <b>12</b>
Protocol	Specify Protocol used	AOS_AINP, AINP
Port	Specify which port the device is connected to the FieldServer	R2

### Example

```
// Server Side Nodes

Nodes
Node_Name , Node_ID , Protocol , Port *
AINP_12 , 12 , AINP , R2
```

## 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
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 provide data for	One of the Node Names specified in Section 6.2
Data_Type	Data type	Master_Param, Master_Cmd, Slave_Param, Slave_Cmd
AINP_Block_Number*	Specify the parameter or command block number 0-29 for Data_Type Master_Param 255 for Data_Type Master_Cmd	<b>0-29, 255</b>
Address*	Specify the first slave command number. This parameter is required only when the Data_Type is Slave_Cmd	<b>0-254</b>
Length	Length of Map Descriptor.	<b>1-255</b>

\* Note that no connection information is necessary on Server side



### 6.3.3 Master Parameter Map Descriptor Example.

The following Map Descriptor allows the driver to accept Master parameters broadcasted by a remote Master device and to send parameter updates to a remote Master device.

```
// Server Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Data_Type , AINP_Block_Number , Length
SMD_Mast_Parm , DA_Mast_Parm , 0 , Passive , AINP_12 , Mast_Param , 1 , 255
```

One of the Data Arrays declared in Section 4 The Driver will store Master parameters for Block 1 in this Data

The location in the Data Array where the first parameter of this block will be stored.

Mast\_Param Data\_Type indicates that this Data Array will store Master Parameters.

This Map Descriptor applies only to Master Parameters for Block 1; Similar Map Descriptors can be created for each block if required.

The length of the Data\_Array must be adequate for the number of parameters in this block.

### 6.3.4 Master Command Map Descriptor Example.

The following Map Descriptor allows the Driver to execute a command at the Master whenever a value is changed on the FieldServer by another Protocol.

```
// Server Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Data_Type , AINP_Block_Number , Length
SMD_Mast_Cmd , DA_Mast_Cmd , 0 , Passive , AINP_12 , Mast_Cmd , 255 , 255
```

One of the Data Arrays declared in Section 4 The Driver will store command data values in this Data Array

The location in the Data Array where the first command data value will be stored.

Mast\_Cmd Data\_Type indicates that this Data Array will store Master Commands.

Block number 255 is reserved for Master commands.

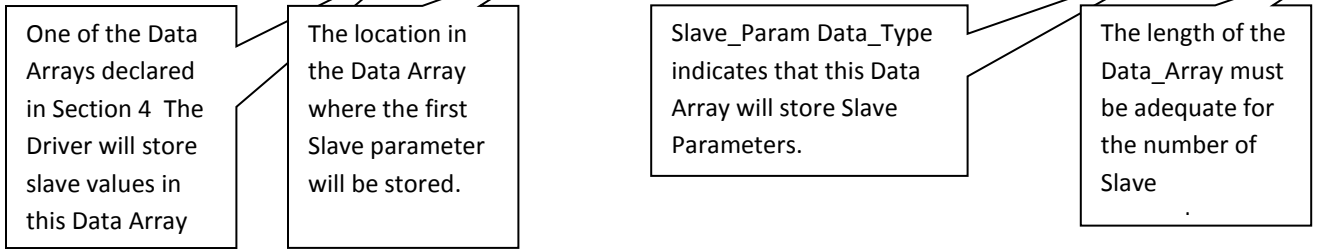
The length of the Data\_Array must be adequate for the number of commands.

### 6.3.5 Slave Parameter Map Descriptor Example.

The following Map Descriptor allows the Driver to store Slave Parameters.

```
// Server Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Data_Type , Length
SMD_Slv_Params , DA_Slv_Param , 0 , Passive , AINP_12 , Slave_Param , 200
```

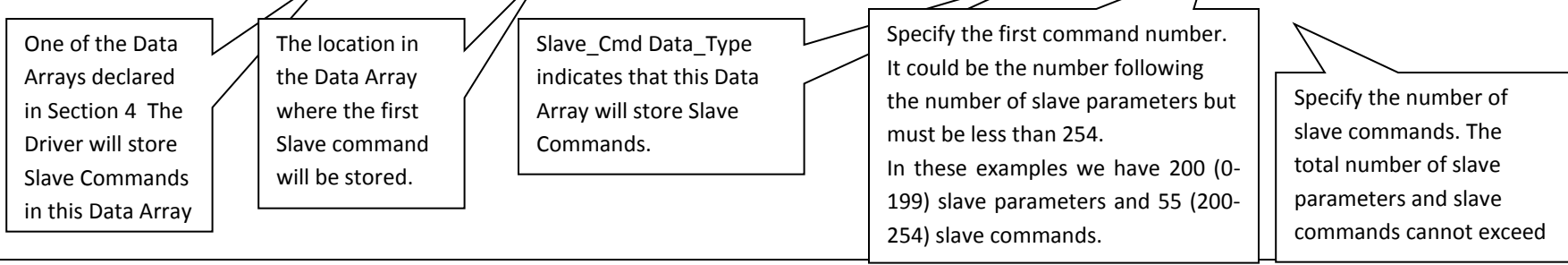


### 6.3.6 Slave Command Map Descriptor Example.

The following Map Descriptor allows the Driver to store Slave Commands. The Master can execute any of the stored commands on the Slave.

```
// Server Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Data_Type , Address , Length
SMD_Slv_Cmd , DA_Slv_Cmd , 0 , Passive , AINP_12 , Slave_Cmd , 200 , 55
```



---

## Appendix A. Useful Features

### Appendix A.1. Parameter 0

Parameter 0 is the only defined parameter and represents the firmware version of the device. The Master parameter 0 will indicate the remote Master controller's firmware version and Slave parameter 0 will indicate the Slave firmware version. The AINP driver version will be used as Slave parameter 0 by the FieldServer if the data value is 0 in the Slave parameter Data Array.