Driver Manual

FS-8700-141 AOS AINP

APPLICABILITY & EFFECTIVITY
Effective for all systems manufactured after March 2021.

Driver Revision: 1.00
Document Revision: 4.A

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 3rd Party Equipment 4

2.2.1 Required 3rd Party Hardware 4

3 Hardware Connections 5

4 Data Array Parameters 6

5 Client Side Configuration 7

6 Server Side Configuration 8

6.1 Server Side Connection Parameters 8

6.2 Server Side Node Parameters 9

6.3 Server Side Map Descriptor Parameters 9

6.3.1 FieldServer Specific Map Descriptor Parameters 9

6.3.2 Driver Specific Map Descriptor Parameters 9

6.4 Map Descriptor Examples 10

6.4.1 Master Parameter 10

6.4.2 Master Command 10

6.4.3 Slave Parameter 11

6.4.4 Slave Command 11

7 Useful Features 12

7.1 Parameter 0 12

1 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 MSA Safety

Part #	Description
FS-8917-16	RJ45 pigtail connector adapter
	Driver manual

2.2 Provided by the Supplier of 3rd Party Equipment

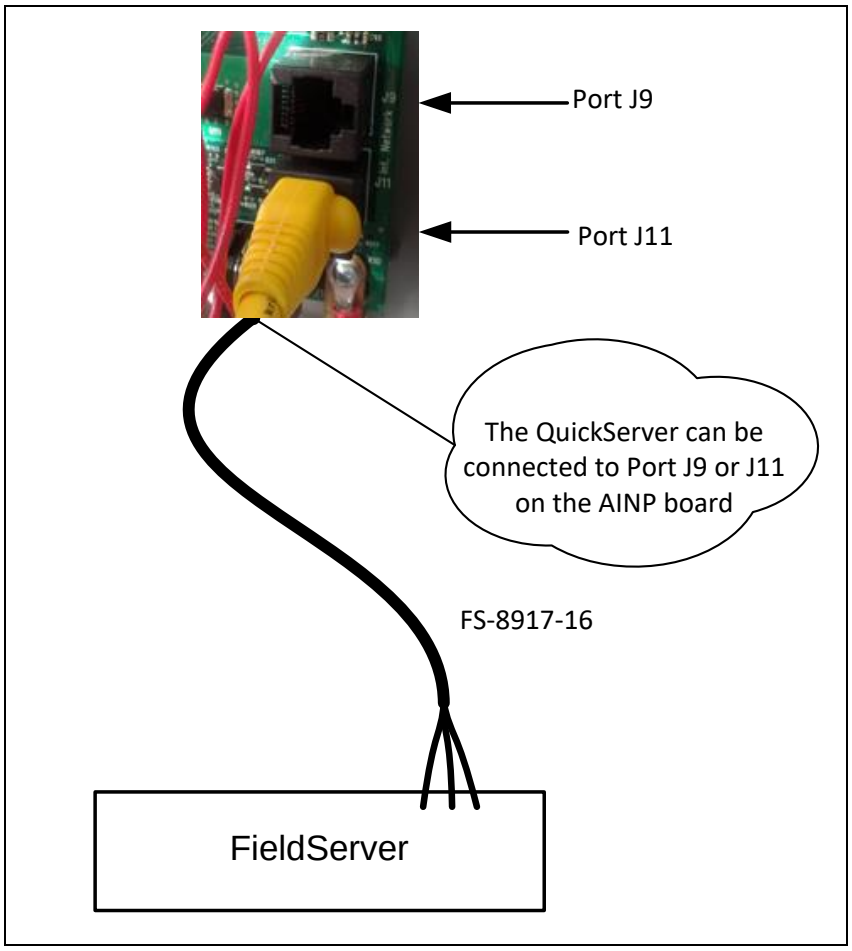
2.2.1 Required 3rd Party Hardware

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#	FieldServer RS-485 Port	Wire 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		
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
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_Mast_Parm       , UInt16               , 255
DA_Mast_Cmd        , UInt16               , 255
DA_Slv_Parm        , UInt16               , 200
DA_Slv_Cmd         , UInt16               , 55
```

5 Client Side Configuration

The Client side of the driver is intended to support MSA Safety'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, contact the FieldServer sales group.

6 Server Side Configuration

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

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with an 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: 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 (with 232/485 converter), R1-R2[1]
Protocol	Specify protocol used.	AOS_AINP, AINP
Baud*	Specify baud rate.	**19200,** 38400 (Protocol limitation)
Parity*	Specify parity.	**None** (Protocol limitation)
Data_Bits*	Specify data bits.	**9**
Stop_Bits*	Specify stop bits.	**1,** 2
Server_Hold_Timeout*	Specifies time FieldServer reserves the Server side connection when waiting for the Client side to update the Data_Array.	>1.0s, **5s**

Example

```
//  Server Side Connections
Connections
Port          , Protocol   , Baud    , Parity   , Data_Bits
R1            , InfraTox   , 19200   , None     , 8
```

[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.

6.2 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, **12**
Protocol	Specify Protocol used.	AOS_AINP, AINP
Port	Specify at which port the device is connected to the FieldServer.	P1-P2[2], R1-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 "Data Array" section above
Data_Array_Offset	Starting location in Data Array.	0 to (Data_Array_Length-1) as specified in "Data_Array" section
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 which data has to be sent.	One of the Node names specified in the "Node" section above
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	**0**-29, 255
Address*	Specify the first slave command number. This parameter is required only when the Data_Type is Slave_Cmd.	**0**-254
Length	Length of Map Descriptor.	**1**-255

[2] Ports P1-P2 require a 232/485 converter.

6.4 Map Descriptor Examples

6.4.1 Master Parameter

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
```

In the above example:

- Data_Array_Name – One of the Data Arrays declared in the Server side Data Array" section. The Driver will store Master parameters for Block 1 in this Data Array.
- Data_Array_Offset – The location in the Data Array where the first parameter of this block will be stored.
- Data_Type – Mast_Param Data_Type indicates that this Data Array will store Master Parameters.
- AINP_Block_Number – This Map Descriptor applies only to Master Parameters for Block 1; Similar Map Descriptors can be created for each block if required.
- Length – The length of the Data_Array must be adequate for the number of parameters in this block.

6.4.2 Master Command

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
```

In the above example:

- Data_Array_Name – One of the Data Arrays declared in the Server side Data Array" section. The Driver will store command data values in this Data Array.
- Data_Array_Offset – The location in the Data Array where the first command data value will be stored.
- Data_Type – Mast_Cmd Data_Type indicates that this Data Array will store Master Commands.
- AINP_Block_Number – Block number 255 is reserved for Master commands.
- Length – The length of the Data_Array must be adequate for the number of commands.

6.4.3 Slave Parameter

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
```

In the above example:

- Data_Array_Name – One of the Data Arrays declared in the Server side **"**Data Array" section. The Driver will store command data values in this Data Array.
- Data_Array_Offset – The location in the Data Array where the first Slave parameter will be stored.
- Data_Type – Slave_Param Data_Type indicates that this Data Array will store Slave Parameters.
- Length – The length of the Data_Array must be adequate for the number of Slave parameters.

6.4.4 Slave Command

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
```

In the above example:

- Data_Array_Name – One of the Data Arrays declared in the Server side **"**Data Array" section. The Driver will store Slave Commands in this Data Array.
- Data_Array_Offset – The location in the Data Array where the first Slave parameter will be stored.
- Data_Type – Slave_Cmd Data_Type indicates that this Data Array will store Slave Commands.
- Address – Specify the first command number. It could be the number following the number of slave parameters but must be less than 254. In these examples we have 200 (0- 199) slave parameters and 55 (200-254) slave commands.
- Length – Specify the number of slave commands. The total number of slave parameters and slave commands cannot exceed 255.

7 Useful Features

7.1 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.