A Sierra Monitor Company

Driver Manual
(Supplement to the FieldServer Instruction Manual)

FS-8700-61 ADC Heatless Communication

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after May 1, 2001

Driver Version:	**1.00**
Document Revision:	**4**

TABLE OF CONTENTS

1. ADC Heatless Communication Description

The Serial ADC Heatless Communication driver allows the Communications FieldServer to transfer data to and from devices over either RS-232 or RS-485 using ADC Heatless Communication protocol. The FieldServer can emulate either a Server or Client.

The Server and Client drivers function according to the following specification:
ADC Heatless Communication TECH-NOTE, Doc no. #1251814, Rev. C.

The Client driver implements the following functionality:
- Polls for Clocks and Counters data.
- Polls for Configuration data.
- Polls for Inputs data.
- Polls for Status data.
- Polls for Tests data.

The Server driver implements the following functionality:
- Provides Clocks and Counters data.
- Provides Configuration data.
- Provides Inputs data.
- Provides Status data.
- Provides Tests data.

The drivers work on a byte for byte protocol. All data poll commands and responses to poll commands are one byte each.

The Server driver implements a response buffer according to the ADC specification. If a response is more than one byte long, the Client polls for the rest of the bytes with special buffer request poll commands. Each poll is still one byte long and produces a one byte reply from the Server driver.

2. Driver Scope of Supply

2.1. Supplied by FieldServer Technologies for this driver

FieldServer Technologies PART #	Description
FS-8917-21	RS-485 Connection adapter
FS-8700-61	Driver Manual.

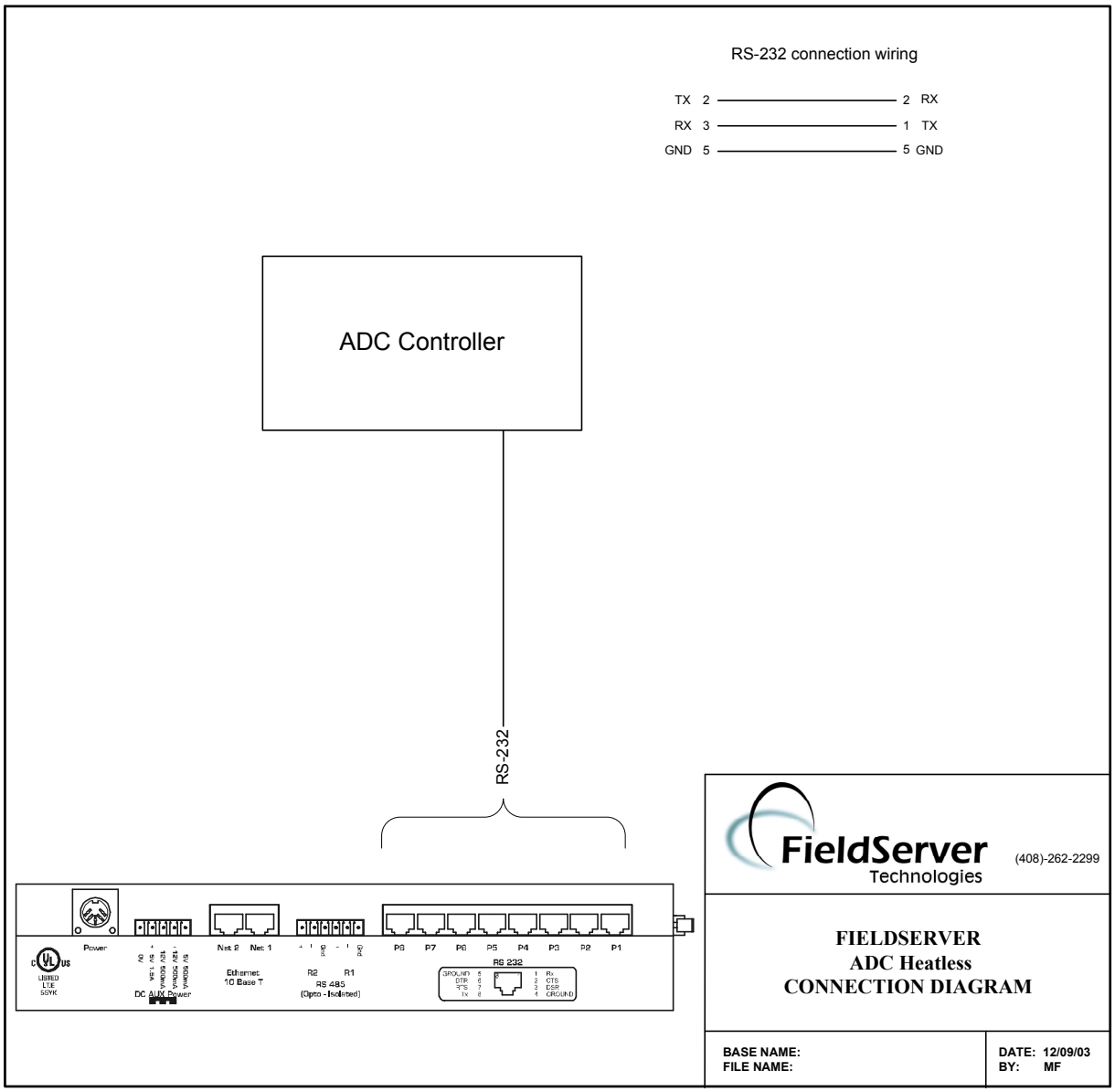
2.2. Provided by user

The user has to provide a RS-232 serial cable for connecting the ADC controller to the FieldServer. The RS-232 cable must contain a DB9 female socket on the end to be connected to the FieldServer. The other end must contain a suitable connector as needed for the ADC controller. The wire has to be a null-modem cable with pins 2 and 3 swapped e.g. Pin 2 from one connector connects to pin 3 of the other.

3. Hardware Connections

The FieldServer is connected to the ADC controller as shown below.

Configure the ADC controller according to manufacturer's instructions. The serial port on the ADC controller is labeled P1 and is located on the base-board of the controller.



4. Configuring the FieldServer as a ADC Heatless Communication 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 an ADC Heatless Communication Server.

4.1. 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 ADC Heatless 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.

4.2. Data Arrays

Section Title		
Data_Arrays		
Column Title	**Function**	**Legal Values**
Data_Array_Name	Provide name for Data Array	Up to 15 alphanumeric characters
Data_Format	Provide data format. Each Data Array can only take on one format.	FLOAT, BIT, UInt16, SInt16, Packed_Bit, Byte, UInt32, Packed_Byte, Swapped_Byte
Data_Array_Length	Number of Data Objects. Must be larger than the data storage area required for the data being placed in this array.	1-10,000

Example

```
//  Data Arrays
//
Data_Arrays
Data_Array_Name,    Data_Format,    Data_Array_Length
Clocks_Counters,    UInt32,         12
Configuration,      UInt16,         8
Inputs,             UInt16,         4
Status,             UInt16,         3
Tests,              Uint16,         24
```

4.3. Client Side Connection Descriptors

Section Title
Connections

Column Title	Function	Legal Values
Port	Specify which port the device is connected to the FieldServer	P1-P8[1]
Baud*	Specify baud rate	300, 1200, 2400 (check ADC controller setup)
Parity*	Specify parity	None
Data_Bits*	Specify data bits	8
Stop_Bits*	Specify stop bits	**1**
Protocol	Specify protocol used	Adc
Handshaking*	Specify hardware handshaking	**None**
Poll Delay*	Time between internal polls	**≥0.2s**

Example

```
// Client Side Connections

Connections
Port,          Baud,    Parity,  Protocol,  Handshaking,  Poll_Delay
P1,            2400,    None,    Adc,       None,         0.200s
```

4.4. 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	Modbus station address of physical Server node	1-255
Protocol	Specify protocol used	Adc
Port	Specify which port the device is connected to the FieldServer	P1-P8[1]

Example

```
// Client Side Nodes

Nodes
Node_Name,     Node_ID,    Protocol,  Port
Node_A,        11,         Adc,       P1
```

[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.

4.5. Client Side Map Descriptors

4.5.1. 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_Location	Starting location in Data Array	0 to maximum specified in “Data Array” section above
Function	Function of Client Map Descriptor	RDBC

4.5.2. 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 “Client Node Descriptor” above
Adc_Command	Data poll commands	Run_Time, Off_Time, Total_Time, Filter_Usage, Valve_Usage, Desiccant_Usage, Filter_Set_Point, Valve_Set_Point, Desiccant_Set_Point, AMLOC_Time, AMLOC_Savings, AMLOC_Perc_Savings, AMLOC_Hold_Point, NEMA_Cycle, Repress_Time, Switching_Failure, AMLOC_Mode, Language, Degree_Units, Dewpoint_Warning, Dewpointer, Left_AMLOC_Probe, Right_AMLOC_Probe, Left_Chamber_Pressure, Right_Chamber_Pressure, Dewpoint, Dryer_Status, Alarm_Warning_Status, Half_Cycle_Time, 00_Test_Pattern, AA_Test_Pattern, FF_Test_Pattern

4.5.3. Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Rate at which data is polled	>= 0.2s

4.5.4. Map Descriptor Example.

```
//  Client Side Map Descriptors

Map Descriptors
Map_Descriptor_Name,  Scan_Interval,  Data_Array_Name,  Data_Array_Offset,  Function,  Node_Name,  Adc_Command
ADC_MBA1,             0.2s    ,       Clocks_Counters,  0,                  Rdbc,      Node_A,     Run_Time
```

This can be any name but each name must be unique. Name will appear in FieldServer Map Descriptor status information screens.

The Data Array name must be one found under Data_Arrays. Data from the scan will be stored into the array at Data_Array_Offset.

Scan interval must be adapted for multiple Map Descriptor scans.

This value specifies the offset into the Data Array (DA_Service_Data) where the data fetched must be stored.

Function may not be Write. Only read, continuous optionally, allowed.

Adc Commands may be one of several commands specified in the table e.g. Run_Time

Node Name must be one found under Nodes, Node_Name. Data will be fetched from this node and port during a scan.

5. Configuring the FieldServer as a ADC Heatless Communication 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 a ADC Heatless Communication Client.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for ADC Heatless Communication 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.

5.1. Server Side Connection Descriptors

Section Title		
Connections		
Column Title	**Function**	**Legal Values**
Port	Specify which port the device is connected to the FieldServer	P1-P8[2]
Baud*	Specify baud rate	300, 1200, **2400** (check ADC controller setup)
Parity*	Specify parity	**None**
Data_Bits*	Specify data bits	**8**
Stop_Bits*	Specify stop bits	**1**
Protocol	Specify protocol used	Adc
Handshaking*	Specify hardware handshaking	**None**

Example

```
// Server Side Connections

Connections
Port,       Baud,     Parity,   Protocol,   Handshaking
P1,         2400,     None,       Adc,      None
```

[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.

5.2. Server Side Node Descriptors

Section Title		
Nodes		
Column Title	**Function**	**Legal Values**
Node_Name	Provide name for node	Up to 32 alphanumeric characters
Node_ID	Modbus station address of physical Server node	1-255
Protocol	Specify protocol used	Adc

Example

```
//  Server Side Nodes

Nodes
Node_Name,          Node_ID,        Protocol
Node_A,             11,             adc
```

5.3. Server Side Map Descriptors

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 “Data Array” section above
Data_Array_Location	Starting location in Data Array	0 to maximum specified in “Data Array” section above
Function	Function of Client Map Descriptor	Passive

5.3.2. 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 “Client Node Descriptor” above

5.3.3. Map Descriptor Example.

```
//  Server Side Map Descriptors

Map Descriptors
Map_Descriptor_Name,    Data_Array_Name,    Data_Array_Offset,    Function,    Node_Name
SMB_AI1,                DA_AI1,             0,                    Passive,     Node_A
```

This can be any name but each name must be unique. Name will appear in FieldServer Map Descriptor status information screens.

The Data Array name must be one found under Data_Arrays. Data from the Forth script file will be stored into the array at Data_Array_Offset. This data will be sent to a requesting Client.

This value specifies the offset into the Data Array (DA_AI1) where the data from the Forth script will be stored. Note that the script can offset the data in addition to this offset value.

Node Name must be one found under Nodes, Node_Name. This defines the Data Array for node name and polls from a Client to this node will be answered with data from this Data Array.

Function may not be read or write since it implements a Server. Function may only be Passive.

THIS PAGE INTENTIONALLY LEFT BLANK