



FieldServer
FS-8700-67 Russelectric Model 2000
Driver Manual
(Supplement to the FieldServer Instruction Manual)

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after November 2015

Kernel Version: 1.04
Document Revision: 1

Contact Information:

Thank you for purchasing the FieldServer.

Please call us for Technical support of the FieldServer product.

Contact Information:

Sierra Monitor Corporation
1991 Tarob Court
Milpitas, CA 95035

Contact number:
+1 408 262-6611
+1 800 727-4377

Email: info@sierramonitor.com

Website: www.sierramonitor.com

TABLE OF CONTENTS

- 1 Russelectric model 2000 Description 4**
- 2 Driver Scope of Supply 5**
 - 2.1 Supplied by Sierra Monitor Corporation for this driver 5
 - 2.2 Provided by the Supplier of 3rd Party Equipment..... 5
 - 2.2.1 *Required 3rd Party Hardware*..... 5
- 3 Hardware Connections..... 6**
- 4 configuring the fieldserver as a russelectric model 2000 client..... 7**
 - 4.1 Data Arrays/Descriptors 7
 - 4.2 Client Side Connection Parameters 8
 - 4.3 Client Side Node Parameters 8
 - 4.4 Client Side Map Descriptor Parameters 9
 - 4.4.1 *FieldServer Specific Map Descriptor Parameters* 9
 - 4.4.2 *Driver Related Map Descriptor Parameters* 9
 - 4.4.3 *Timing Parameters* 9
 - 4.5 Map Descriptor Example. 10
- 5 Configuring the FieldServer as a russelectric Model 2000 Server 11**
 - 5.1 Server Side Connection Descriptors 11
 - 5.2 Server Side Node Descriptors 12
 - 5.3 Server Side Map Descriptors..... 12
 - 5.3.1 *FieldServer Specific Map Descriptor Parameters* 12
 - 5.3.2 *Driver Specific Map Descriptor Parameters* 12
 - 5.3.3 *Map Descriptor Example.* 13

LIST OF FIGURES

- Figure 1 - Generic Connection Diagram.....6

1 RUSSELECTRIC MODEL 2000 DESCRIPTION

The Serial Russelectric Model 2000 driver allows the FieldServer to transfer data to and from devices over either RS-232 or RS-485 using RTU protocol. The Russelectric Model 2000 drivers implement a Model 2000 Client and a Model 2000 Server. The Client driver can read data from a remote Server and send write data commands. The Server driver emulates a Model 2000 device and responds to data read and write poll commands.

Messages

The client driver implements the following RTU message or command types:

- Read_Output_Table – Used to read the bit values of the discrete outputs on the Model 2000.
- Read_Registers – Used to read the integer values of the Model 2000's internal registers.
- Force_Single_Output – Used to set or clear the bit value of a single discrete output on the Model 2000.
- Preset_Single_Register – Used to load an integer value into a specific single register of the Model 2000.

Addressing

Every Model 2000 has a unique address assigned to it. The client driver can send messages to a specific Model 2000 by using the unique address in the message or by using a special broadcast address of value 0. Only the following messages can be sent in broadcast by the client (other messages are read data types and would cause a communications collision when all Model 2000 devices respond):

- Force_Single_Output
- Preset_Single_Register

All Model 2000 devices have to receive and process a broadcast message. No device may send a response message in reply. Subsequent data read commands by the client driver will update the data that was changed or loaded by a broadcast message

2 DRIVER SCOPE OF SUPPLY

2.1 Supplied by Sierra Monitor Corporation for this driver

Sierra Monitor Corporation PART #	Description
FS-8917-02	RJ45 to DB9F connector adapter
FS-8700-67	Driver Manual

2.2 Provided by the Supplier of 3rd Party Equipment

2.2.1 Required 3rd Party Hardware

The user has to provide a suitable cable with appropriate connectors that will connect the Model 2000 either directly to the FieldServer or to the RS-232 to RS-485 converter. When connecting to a converter, consult the converter's documentation for pinouts.

3 HARDWARE CONNECTIONS

The FieldServer is connected to the Model 2000 as shown below.

Configure the Model 2000 according to manufacturer's instructions.

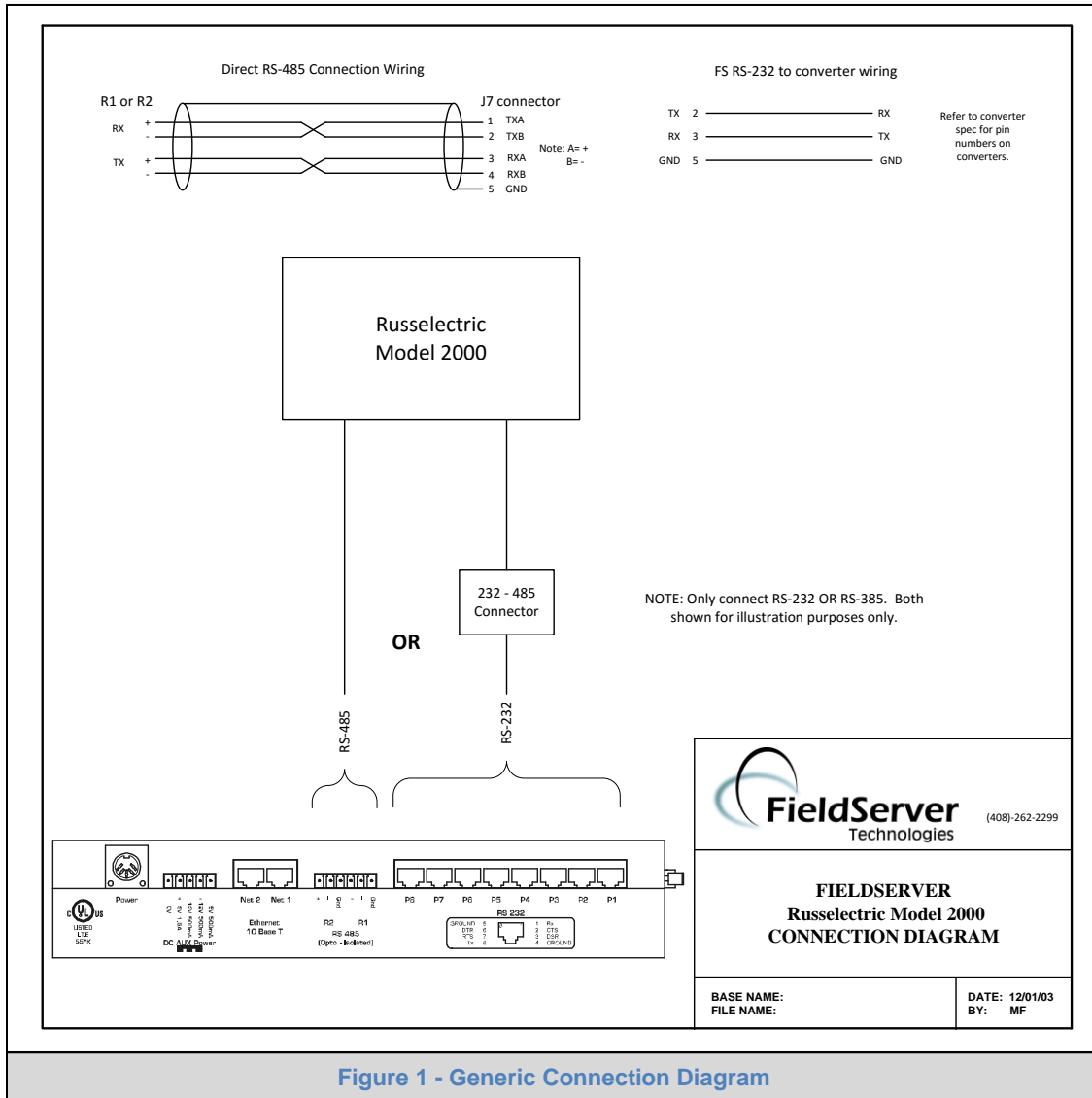


Figure 1 - Generic Connection Diagram

4 CONFIGURING THE FIELD SERVER AS A RUSSELECTRIC MODEL 2000 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 a Russelectric Model 2000 Server.

4.1 Data Arrays/Descriptors

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, Uint16, Sint16, Backed_Bit, Byte, Packed_byte, Swapped_Byte
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_Format , Data Array Length
RTU_Registers , Uint16 , 79
RTU_Digital_Out , Bit , 34
```

4.2 Client Side Connection Parameters

Section Title		
Connections		
Column Title	Function	Legal Values
Port	Specify which port the device is connected to the FieldServer	P1-P8, R1-R2 ¹
Protocol	Specify protocol used	Rus, RussElectric, Model 2000
Baud*	Specify baud rate	110-115200 Standard baud rates only (set to same value as used on Model 2000)
Parity*	Specify parity	Even, Odd, None, Mark, Space (Refer to Model 2000 set up)
Data_Bits*	Specify data bits	8
Stop_Bits*	Specify stop bits	1
Poll_Delay*	Time between internal polls	0.5 seconds

Example

```
// Client Side Connections

Connections
Port , Protocol , Baud , Parity , Poll_Delay
P1 , Rus , 9600 , None , 1.0 s
```

4.3 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	Model 2000 station address	0-247 (Node 0 used exclusively to broadcast)
Protocol	Specify Protocol used	Rus
Port	Specify through which port the device is connected to the FieldServer	P1-P8, R1-R2 ²

Example

```
// Client Side Nodes

Nodes
Node_Name , Node_ID, , Protocol , Port
Node_1 , 1 , Rus , 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.

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.

4.4 Client Side Map Descriptor Parameters

4.4.1 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 maximum as specified in Data Array section
Function	Function of Client Map Descriptor	Rdbc, Wrbc, Wrbx

4.4.2 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. Node_0 can be used to broadcast a command. Only write functions can be used with Node_0
Data_Type	Used to indicate access to the Model 2000 Registers or Points (flags)	Register Flag
Length	The number of registers or points	For write functions, length defaults to 1. Length must be less than or equal to the maximum number of points or registers available. See your hardware manual for details
Address	The starting register or point number	1-10,000 1-10,000

4.4.3 Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Rate at which data is polled	≥0.5 s

4.5 Map Descriptor Example.

Map_Descriptor_Name	Scan_Interval	Data_Array_Name	Data_Array_Offset	Function	Node_Name	Data_Type	Address	Length
RUS_MBA1	, 3.0s	, RTU_Digital_Out	, 0	, Rdbc	, Node_1	, Flag	, 1	, 34

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

This value specifies the offset into the data array (RTU_Digital_Out) where the data fetched must be stored.

The function may be read for read commands and write for write type commands. Continuous operations are allowed.

Use Register to access Model 2000 registers or Flag to access the output table

Scan interval must be adapted for multiple Map Descriptor scans to prevent a situation where not all Map Descriptors can be executed in a certain time. Take remote device response time into consideration here.

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.

Node name must be one found under Nodes, Node_name. Data will be fetched from this node and port during a scan. Use Node_0 with id of 0 to execute a broadcast command.

5 CONFIGURING THE FIELD SERVER AS A RUSSELECTRIC MODEL 2000 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 Russelectric Model 2000 Client

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for Russelectric Model 2000 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, R1-R23
Protocol	Specify protocol used	Rus
Baud*	Specify baud rate	110 – 115200, standard baud
Parity*	Specify parity	Even, Odd, None, Mark, Space (refer to Model 2000 setup)
Data_Bits*	Specify data bits	8
Stop_Bits*	Specify stop bits	1
Handshaking*	Specify hardware handshaking	None

Example

```
// Server Side Connections

Connections
Port      , Baud      , Parity    , Protocol
P1       , 9600      , None     , Rus
```

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.

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	Model 2000 station address to emulate	0-247 (Node 0 used exclusively to receive broadcast messages)
Protocol	Specify protocol used	Rus

Example

```
// Server Side Nodes

Nodes
Node_Name      , Node_ID      , Protocol
Node_1         , 1           , Rus
```

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_Offset	Starting location in Data Array	0 to maximum specified in "Data Array" section above
Function	Function of Server Map Descriptor	Server

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
Data_Type	Used to emulate Model2000 Registers or Points (flags)	Register Flag
Address	The start register number or start point number	1-10,000
Length	The number of registers or points	1-10,000

5.3.3 Map Descriptor Example.

Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_Name	Data_Type	Address	Length
RUS_SMB1	RTU_Digital_Out	0	Server	Node_1	Flag	1	34

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.

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

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.

Use Register or Flag to emulate Model 2000 Registers or Digital Output Points.