

Driver Manual

FS-8704-21 Cresnet/IP

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after June 2023.



Driver Revision: 1.02
Document Revision: 3.B



fieldserver

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	4
2	Driver Scope of Supply	4
	2.1 Supplied by MSA Safety.....	4
	2.2 Provided by the Supplier of 3 rd Party Equipment	4
3	Hardware Connections	5
	3.1 Connection Notes	5
4	Data Array Parameters	6
5	Client Side Configuration	7
	5.1 Client Side Connection Descriptions	7
	5.2 Client Side Node Descriptors	7
	5.3 Client Side Map Descriptors	8
	5.3.1 FieldServer Related Map Descriptor Parameters	8
	5.3.2 Driver Related Map Descriptor Parameters	8
	5.3.3 Timing Parameters	8
	5.4 Map Descriptor Example	9
6	Server Side Configuration	10
	6.1 Server Side Connection Parameters	10
	6.2 Server Side Node Parameters.....	10
	6.3 Server Side Map Descriptor Parameters.....	11
	6.3.1 FieldServer Specific Map Descriptor Parameters	11
	6.3.2 Driver Specific Map Descriptor Parameters	11
	6.4 Map Descriptor Example	11

1 Description

The Cresnet/IP driver allows the FieldServer to transfer data to and from devices over Ethernet using Crestron TCP_CIP protocol. This driver allows the FieldServer to communicate with Crestron Control systems or any 3rd party device supporting Crestron TCP_CIP protocol. Crestron Control systems are gateways to various other Crestron or other devices such as touch panels, keypads lighting controls etc.

This driver is based upon Crestron's TCP_CIP Protocol Specification (OEM Version 1.17 10/6/06) and Cresnet II packet Formats 11/18/2008.

The FieldServer can be configured as a Client or Server.

Max Nodes Supported

FieldServer Mode	Nodes ¹	Comments
Client	252	Crestron Control systems can control a maximum of 252 devices.
Server	255	FieldServer can emulate up to 255 Nodes.

2 Driver Scope of Supply

2.1 Supplied by MSA Safety

Part #	Description
	Driver Manual

2.2 Provided by the Supplier of 3rd Party Equipment

Part #	Description
	Ethernet 10/100 BaseT Switch ²
Crestron Control System (CP2E)	Control System with Ethernet adapter supporting Crestron's TCP_CIP protocol

¹ This is further restricted to the maximum number of sockets on any FieldServer model.

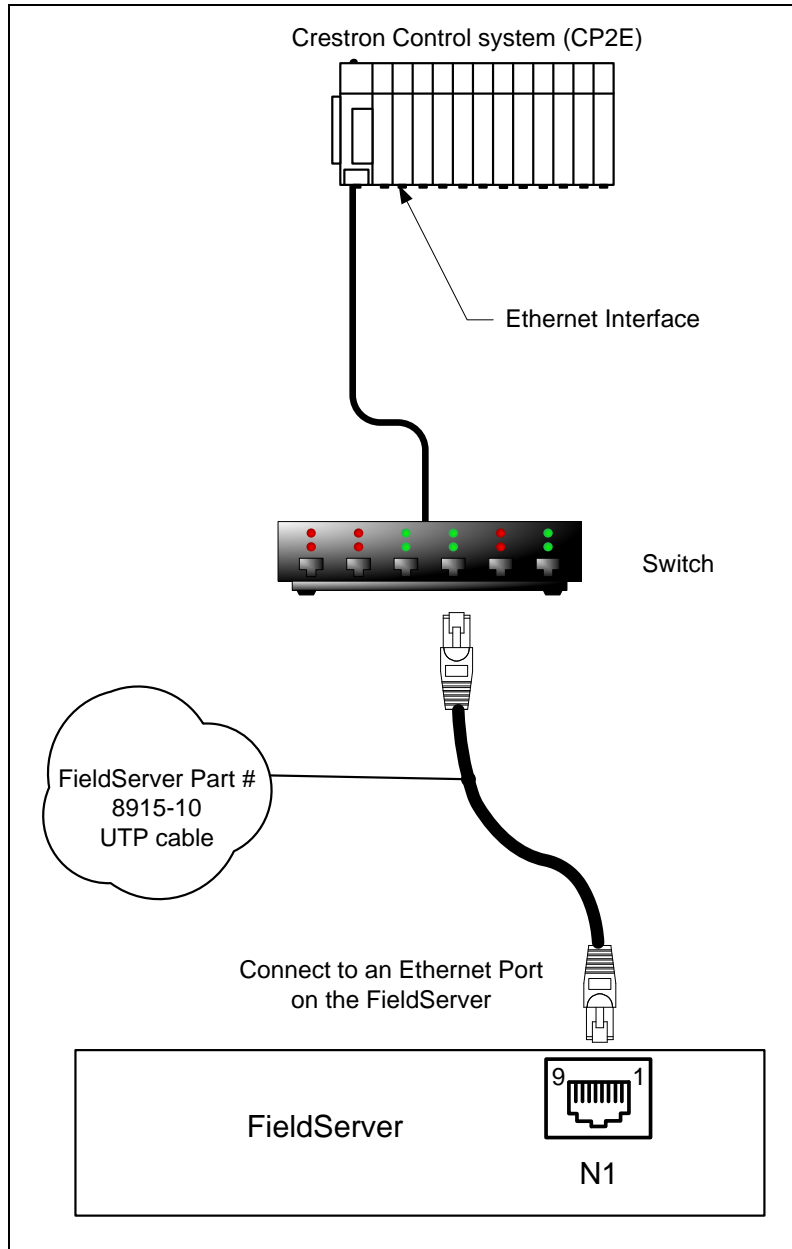
² Not all FieldServer models support 100BaseT. Consult the appropriate instruction manual for details of the Ethernet speed supported by specific hardware.

3 Hardware Connections

The FieldServer is connected to the Crestron Control system (e.g. CP2E) as shown in connection drawing.

It is possible to connect a Crestron Control system Ethernet device using the N1 or the N2 network ports. These ports just need to be configured for Cresnet/IP in the configuration file.

Configure Crestron Control system according to manufacturer's instructions.



3.1 Connection Notes

Cresnet/IP Servers should use TCP port 41794.

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_AI_01 , Float , 200
DA_AO_01 , Float64 , 200
DA_DI_01 , Bit , 200
DA_DO_01 , Bit , 200
```

5 Client Side Configuration

For detailed information on 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 a Cresnet/IP Server.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for Cresnet/IP 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: In the tables below, * indicates an optional parameter and bold legal values are defaults.

5.1 Client Side Connection Descriptions

Section Title		
Connections		
Column Title	Function	Legal Values
Adapter	Adapter Name	N1-N2, WLAN ³
Protocol	Specify protocol used.	Cresnet/IP

Example

```
// Client Side Connections
Connections
Adapter , Protocol
N1 , Cresnet/IP
```

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	IP_ID of the target device. Use Crestron tools to find the IP_ID in the Controller's IP_Table.	3-254
Protocol	Specify protocol used.	Cresnet/IP, Cresnet
Adapter	Specify on which port the device is connected to the FieldServer.	N1-N2, WLAN ³
IP_Address	IP address of the remote Crestron device.	Decimal representation of IP Address

Example

```
// Consumer (Passive Client) Side Nodes
Nodes
Node_Name , Node_ID, , Protocol , Connection , IP_Address
Crestron_01 , 4 , Cresnet/IP , N1 , 192.168.1.174
```

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.

5.3 Client Side Map Descriptors

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 to check heartbeat of target device. If Passive; driver can accept join updates. If Wr bx; driver can send join updates.	Rdbc, Passive, Wr bx

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 the Node section
Length*	Number of continuous joins starting with the join number as specified under "Address" parameter.	1, 2, 3 ... etc.
Data_Type*	Specify Flag for Digital join, Register for Analog join, – for Rdbc.	Flag, Register, -
Address*	Starting join number of this block of joins.	0, 1, 2, 3 ... etc.

5.3.3 Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Rate at which HeartBeat request will be send to target device	≥0.001s

5.4 Map Descriptor Example

One Rdbc Map Descriptor is required per Node to start communication and to check the heartbeat of the target device.

Passive Map Descriptors are required to accept and store join updates from the target device.

Wrbx Map Descriptors are required to send join updates to the target device.

```
// Client Side Map Descriptors
```

Map_Descriptors								
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_Name	Data_Type	Address	Length	Scan_Interval
CMD_HeatBeat	HeartBeat	,0	,Rdbc	,Cresnet_01	,-	,0	,1	,10s
CMD_Analog_01	,DA_Ana_01	,0	,Passive	,Cresnet_01	,Register	,0	,20	,-
CMD_Digital_01	,DA_Dig_01	,0	,Passive	,Cresnet_01	,Flag	,0	,20	,-
CMD_Analog_01_W	,DA_Ana_01_W	,0	,Wrbx	,Cresnet_01	,Register	,0	,20	,-
CMD_Digital_01_W	,DA_Dig_01_W	,0	,Wrbx	,Cresnet_01	,Flag	,0	,20	,-

In the above example:

- **Data_Array_Name** – Data processed by this Map Descriptor is stored in this array. The Data Array holds the data fetched from target device. The HeartBeat Data Array value will be incremented each time the FieldServer receives a HeartBeat response.
- **Function** – Rdbc (read continuously) is used to check the HeartBeat of the target device as well as for startup data synchronization. Passive Map Descriptors store analog or digital join data. When another protocol (e.g. Modbus) updates a location in a Data Array associated with Wrbx, a corresponding join update will be sent to the target device.
- **Node_Name** – One of the Nodes defined in the Node section.

6 Server Side Configuration

For detailed information on 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).

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for Cresnet/IP 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 must 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
Adapter	Adapter Name.	N1-N2, WLAN ⁴
Protocol	Specify protocol used.	Cresnet/IP
CRNT_Server*	Allows the user to specify the device is emulating a Cresnet Server. Normally, the Client starts communication and the Server listens for communication. In the same context CRNT_SERVER specifies if the FieldServer should act as a Client or Server. NOTE: If an error occurs, create an FS Toolbox capture with the CRNT_SERVER set to “Yes”.	Yes, -

Example

```
// Server Side Connections
Adapters
Adapter      , Protocol      , CRNT_Server
N1           , Cresnet/IP     , Yes
```

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	IP_ID that should be used by Client to connect.	1-255
Protocol	Specify protocol used.	Cresnet/IP
Server_Hold_Timeout*	Specifies the time that the FieldServer will reserve the Server side connection while waiting for the Client side to update data in the Data_Array.	>1.0s, 5s
Adapter	Provide Adapter name.	N1-N2, WLAN ⁴

Example

```
// Producer(Active Server) Side Nodes
Nodes
Node_Name           , Node_ID   , Protocol   , Adapter
Cres_Srv_01        , 4         , Cresnet/IP , N1
```

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.

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. If Passive; driver can accept join updates. If Wr bx; driver can send join updates.	Passive, Wr bx

6.3.2 Driver Specific Map Descriptor Parameters

Column Title	Function	Legal Values
Node_Name	Name of Node to which data is sent.	One of the Node names specified in the Server "Node" section above
Length*	Number of continuous joins starting with the join number as specified under "Address" parameter.	1, 2, 3 ... etc.
Data_Type*	Data type.	Flag, Register, -
Address*	Starting join number of this block of joins.	0, 1, 2, 3 ... etc.

6.4 Map Descriptor Example

Individual Map Descriptors need to be created to provide/accept analog or digital join update information. Passive Map Descriptors are used to accept join updates, Wr bx Map Descriptors are used to send join updates.

```
// Server Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Data_Type , Address , Length
SMD_Ana_01 , DA_Ana_01 , 0 , Passive , Cresnet_01 , Register , 0 , 20
SMD_Dig_01 , DA_Dig_01 , 0 , Passive , Cresnet_01 , Flag , 0 , 20
SMD_Ana_01_W , DA_Ana_01_W , 0 , Wr bx , Cresnet_01 , Register , 0 , 1
SMD_Dig_01_W , DA_Dig_01_W , 0 , Wr bx , Cresnet_01 , Flag , 0 , 1
```

In the above example:

- Data_Array_Offset – The Data Array where the join data will be stored.
- Function – Passive Map Descriptors store analog or digital join data. When another protocol (e.g. Modbus) updates a location in a Data Array associated with Wr bx, a corresponding join update will be sent to the target device.
- Node_Name – One of the Nodes defined in the Node section.