Driver Manual
(Supplement to the FieldServer Instruction Manual)

FS-8705-10
Barrington MicroStar Serial Driver

APPLICABILITY & EFFECTIVITY
Effective for all systems manufactured after Aug 20, 2012
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TABLE OF CONTENTS</strong></td>
<td>2</td>
</tr>
<tr>
<td>1 Barrington MicroStar Driver Description</td>
<td>4</td>
</tr>
<tr>
<td>2 Driver Scope of Supply</td>
<td>5</td>
</tr>
<tr>
<td>2.1 Supplied with this driver</td>
<td>5</td>
</tr>
<tr>
<td>3 Hardware Connections</td>
<td>6</td>
</tr>
<tr>
<td>3.1 Hardware Connections</td>
<td>6</td>
</tr>
<tr>
<td>3.2 Block Diagram</td>
<td>8</td>
</tr>
<tr>
<td>4 Configuring the FieldServer as a Barrington MicroStar Client</td>
<td>9</td>
</tr>
<tr>
<td>4.1 Data Arrays</td>
<td>10</td>
</tr>
<tr>
<td>4.1.1 Data Arrays - Example</td>
<td>10</td>
</tr>
<tr>
<td>4.2 Client Side Connections</td>
<td>11</td>
</tr>
<tr>
<td>4.2.1 Client Side Connection Descriptions - Example</td>
<td>12</td>
</tr>
<tr>
<td>4.3 Client Side Nodes</td>
<td>13</td>
</tr>
<tr>
<td>4.3.1 Client Side Nodes - Example</td>
<td>13</td>
</tr>
<tr>
<td>4.4 Client Side Map Descriptors</td>
<td>14</td>
</tr>
<tr>
<td>4.4.1 FieldServer Related Map Descriptor Parameters</td>
<td>14</td>
</tr>
<tr>
<td>4.4.2 Driver Related Map Descriptor Parameters</td>
<td>15</td>
</tr>
<tr>
<td>4.5 Examples</td>
<td>16</td>
</tr>
<tr>
<td>4.5.1 Map Descriptor Example 1 – Read Points</td>
<td>16</td>
</tr>
<tr>
<td>5 Sample Configuration File</td>
<td>17</td>
</tr>
<tr>
<td>6 Configuring the FieldServer as a Barrington MicroStar Server</td>
<td>18</td>
</tr>
<tr>
<td>Appendix 1.1. Scaling</td>
<td>20</td>
</tr>
</tbody>
</table>

© Chipkin Automation Systems, 3495 Cambie St. Unit 211, Vancouver, BC, Canada, VSZ 4R3
Tel: (866) 383-1657, Fax: (416) 915-4024
Appendix 1.2. Supported Communications functions ................................................................. 21
Appendix 1.3. Revision History .................................................................................................... 22
1 Barrington MicroStar Driver Description

This driver is used to exchange data between a FieldServer and a ‘MicroSTAR’ devices. They are also commonly known as ‘Barrington MicroSTAR’ devices. For this driver to work, the LanStar must be removed as the gateway becomes the master on the trunk of Microstars.

Each device has 4x AO, 4x AI, 4x DI, 4x DO and 4x Counters. The devices can be daisy chained on a RS485 network. Setpoints, virtual Digital Outputs, any virtual points or virtual Microstars all reside in the memory of the LanStar and are not accessible using this driver.

The driver can monitor the status of these points and write to the outputs.

This is an active client driver. The FieldServer is the client. The MicroSTAR devices are the passive servers. The Client sends messages and processes responses from the MicroSTAR’s.

The driver is a serial driver using a RS485 serial ports to connect between the FieldServer and the MicroSTAR’s. If RS232 ports are used then a converter to RS485 must be used.

The driver is fully compatible with other FieldServer drivers and meets FieldServer’s quality assurance standards. The driver was developed by Chipkin Automation Systems, an Approved FieldServer Integrator.

Both Client and Server functionality have been implemented. This means that you can use the driver as a client to monitor/command the devices or you can use it a server to make another type of device appear as if it were a MicroSTAR device. Ie. The server functionality allows you to make a foreign device emulate a MicroSTAR so it can be installed in a MicroSTAR trunk.

<table>
<thead>
<tr>
<th>FIELDSERVER MODE</th>
<th>NODES</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTIVE CLIENT</td>
<td>16</td>
<td>A MAXIMUM OF 16 DEVICES MAY BE CONNECTED PER TRUNK. THIS IS A LIMITATION OF THE PROTOCOL WHICH ONLY ALLOWS 16 POSSIBLE ADDRESSES.</td>
</tr>
</tbody>
</table>

Formal Driver Type – Serial - Active Client
2 Driver Scope of Supply

2.1 Supplied with this driver

<table>
<thead>
<tr>
<th>FieldServer Technologies</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PART #</td>
<td></td>
</tr>
<tr>
<td>8915-10</td>
<td>No specific cables are shipped with this driver. A generic RJ45 Ethernet cable is shipped with the hardware.</td>
</tr>
</tbody>
</table>
3 Hardware Connections

Multiple WorkStation protocols and connection supported. See list of FieldServer Drivers.

3.1 Hardware Connections
QuickServer Connections

The other end of this cable connect to RS485 port on the Microstar. That connection is shown in other images.
3.2 Block Diagram

FS30 shown as representing FS20, FS30 and FS40

- BACnet IP
- BACnet Ethernet
- MODBUS TCP
- Rockwell Ethernet/IP or CSP
- HTTP and other web protocols
- GE-SRTP or GE-EGD
- Lonworks Device
- Lonworks Network
- RS485
- MicroStar
- MicroStar
- MicroStar
- Other serial protocols such as Rockwell DF1, GE-SNP, JCI Metasys N2, Bacnet MSTP, Modbus, Ascii
- We are always adding to the library of protocols and can add yours.

FS30 Series shown
Also supported on QuickServer and other platforms.
4 Configuring the FieldServer as a Barrington MicroStar 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 FS).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with a Barrington MicroStar system.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for Barrington MicroStar monitoring, 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.
### 4.1 Data Arrays

<table>
<thead>
<tr>
<th>Section Title</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Data_Arrays</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Column Title</th>
<th>Function</th>
<th>Legal Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data_Array_Name</td>
<td>Provide name for Data Array</td>
<td>Up to 15 alphanumeric characters</td>
</tr>
<tr>
<td>Data_Array_Format</td>
<td>Provide data format. Each Data Array can only take on one format.</td>
<td><strong>Recommended</strong>: Bit, UInt16, Also Supported: Float, UInt32, SInt16, Packed_Bit, Byte, Packed_Bit, Byte, Swapped_Bit</td>
</tr>
<tr>
<td>Data_Array_Length</td>
<td>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.</td>
<td>1-10,000</td>
</tr>
</tbody>
</table>

#### 4.1.1 Data Arrays - Example

```plaintext
// Data Arrays
Data_Arrays
Data_Array_Name, Data_Format, Data_Array_Length,
MicroStarStats, UNT16, 200
```
### 4.2 Client Side Connections

Create one connection for each Barrington MicroStar serial port. Each connection can only be used to connect to a single Barrington MicroStar interface/port.

<table>
<thead>
<tr>
<th>Section Title</th>
<th>Column Title</th>
<th>Function</th>
<th>Legal Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connections</td>
<td>Port</td>
<td>Specify which port the device is connected to the FieldServer</td>
<td>P1-P8</td>
</tr>
<tr>
<td></td>
<td>Protocol</td>
<td>Specify protocol used</td>
<td>Barrington MicroStar</td>
</tr>
<tr>
<td></td>
<td>Baud*</td>
<td>Specify baud rate</td>
<td>Driver Supports: 110; 300; 600; 1200; 2400; 4800; <strong>9600</strong>; 19200; 28800; 38400; 57600 Baud</td>
</tr>
<tr>
<td></td>
<td>Data_Bits *</td>
<td>Specify parity</td>
<td>Driver Supports: 7,8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Barrington MicroStar supports: 7</strong></td>
</tr>
<tr>
<td></td>
<td>Stop_Bits*</td>
<td>Specify data bits</td>
<td>Driver Supports: 1,2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Barrington MicroStar supports: 1</strong></td>
</tr>
<tr>
<td></td>
<td>Parity *</td>
<td>Specify stop bits</td>
<td>Driver Supports: Odd, Even, <strong>None</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Barrington MicroStar supports: Even</strong></td>
</tr>
<tr>
<td></td>
<td>Mstar_checksum_offset</td>
<td>In forming a poll to the Microstar, the driver starts computing the checksum at the 3rd byte. Ie buffer[2] and hence the value 2 is used.</td>
<td>Value should be set to 2 unless directed by CAS support</td>
</tr>
<tr>
<td></td>
<td>Mstar_data_offset</td>
<td>Driver expects the data to start</td>
<td>Value should be set to 0 unless</td>
</tr>
</tbody>
</table>
in the 4th byte of a response. Adjust this expectation by adjusting the offset.

directed by CAS support

| Mstar_invert_data | When set to 1 the byte order in the 16 bit word is reversed in computing the analog value. | Value should be set to 0 unless directed by CAS support |

4.2.1 Client Side Connection Descriptions - Example

```c
// Client Side Connections

Connections

Port, Baud, Parity, Protocol, Data_Bits, Stop_Bits

P1, 9600, Even, Barrington MicroStar, 7, 1
```
4.3 Client Side Nodes

Create one Node per FACP in the network only.

<table>
<thead>
<tr>
<th>Section Title</th>
<th>Nodes</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Column Title</th>
<th>Function</th>
<th>Legal Values</th>
</tr>
</thead>
</table>
| Node_Name    | Provide name for node | Up to 32 alphanumeric characters  
|              |           | NB! The name does not need to correspond to the Node name configured in the Barrington MicroStar system. |
| Node_ID      | Station address of physical server node  
|              | This parameter is not used directly by the driver. We recommend that a unique Node ID’s be given to each node. | 0-64  
|              | Corresponds to the Node numbers of panels. Master node is typically zero. |
| Protocol     | Specify protocol used | Barrington MicroStar |

4.3.1 Client Side Nodes - Example

```
// Client Side Nodes

Nodes

Node_Name, Node_ID, Protocol, Connection
MainPanel, 0, Barrington MicroStar, P1
```
### 4.4 Client Side Map Descriptors

#### 4.4.1 FieldServer Related Map Descriptor Parameters

<table>
<thead>
<tr>
<th>Column Title</th>
<th>Function</th>
<th>Legal Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Map_Descriptor_Name</td>
<td>Name of this Map Descriptor</td>
<td>Up to 32 alphanumeric characters</td>
</tr>
<tr>
<td>Data_Array_Name</td>
<td>Name of Data Array where data is to be stored in the FieldServer</td>
<td>One of the Data Array names from “Data Array” section above</td>
</tr>
<tr>
<td>Data_Array_Offset</td>
<td>Starting location in Data Array</td>
<td>0 to maximum specified in “Data Array” section above</td>
</tr>
<tr>
<td>Function</td>
<td>Function of Client Map Descriptor..</td>
<td>Passive</td>
</tr>
</tbody>
</table>
## 4.4.2 Driver Related Map Descriptor Parameters

<table>
<thead>
<tr>
<th>Column Title</th>
<th>Function</th>
<th>Legal Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node_Name</td>
<td>Name of Node to fetch data from</td>
<td>One of the node names specified in “Client Node Descriptor” above</td>
</tr>
<tr>
<td>Data_Type</td>
<td>This commonly used parameter is not used by this driver.</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>Length of Map Descriptor</td>
<td>Always set to 14 as 14 points are served from each Microstar device.</td>
</tr>
<tr>
<td></td>
<td>Reserves space in the Data Array.</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>The Point Number</td>
<td>Always set to zero.</td>
</tr>
</tbody>
</table>
4.5 Examples

4.5.1 Map Descriptor Example 1 – Read Points

In this example points are reach from 2 nodes

```
Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Node_Name , Function , Length , Address
Read_Node0   , DA_MSTAR00   , 0       , MSTAR00   , rdbc    , 14   , 1
Read_Node1   , DA_MSTAR01   , 0       , MSTAR01   , rdbc    , 14   , 1
```

Data Array and offset where data will be stored.

Node Name of a Node defined in the Nodes section. Node Number must correspond to the Barrington Lanstar panel number. Name must match too.

Read continuously, over and over. Use Scan_Interval to control frequency.

Set Length to 14

Descriptive only. Not used.
5 Sample Configuration File

```plaintext
// Common Information

Bridge
Title ,system_Node_id
2NodeSample ,3

// Data Arrays

Data_Arrays
Data_Array_Name ,Data_Format ,Data_Array_Length
DA_Node_01 ,Float ,20
DA_Node_02 ,Float ,20

// Client Side Connections

Connections
Port ,Baud ,Data_Bits ,Stop_Bits ,Parity ,Protocol
R1 ,9600 ,8 ,1 None ,STAR

// Client Side Nodes

Nodes
Node_Name ,Node_ID ,Protocol
Node_01 ,1 ,STAR
Node_02 ,1 ,STAR

// Client Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name ,Scan_Interval ,Data_Array_Name ,Data_Array_Offset ,Function ,Node_Name ,Length
ReadNode1 ,0s ,DA_Node_01 ,0 ,Rdbc ,Node_01 ,14
ReadNode2 ,0s ,DA_Node_01 ,1 ,Rdbc ,Node_01 ,14
```

---

5 Sample Configuration File

```plaintext
// Common Information

Bridge
Title ,system_Node_id
2NodeSample ,3

// Data Arrays

Data_Arrays
Data_Array_Name ,Data_Format ,Data_Array_Length
DA_Node_01 ,Float ,20
DA_Node_02 ,Float ,20

// Client Side Connections

Connections
Port ,Baud ,Protocol ,Timeout ,Connection_Type ,Max_Master
R2 ,38400 ,BACnet_MSTP ,30 MSTP_Master_Mode ,127

// Client Side Nodes

Nodes
Node_Name ,Node_ID ,Protocol
Bac_Node_01 ,01 ,BACnet_MSTP
Bac_Node_02 ,02 ,BACnet_MSTP

// Client Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name ,Data_Array_Name ,Data_Array_Offset ,Function ,Node_Name ,Object_Id ,Data_Type ,Data_Array_Low_Scale ,Data_Array_High_Scale ,Node_Low_Scale ,Node_High_Scale
AI_01 ,DA_Node_01 ,0 ,Server ,Bac_Node_01 ,1 ,AI ,0 100
AI_02 ,DA_Node_01 ,1 ,Server ,Bac_Node_01 ,2 ,AI ,0 100
AI_03 ,DA_Node_01 ,2 ,Server ,Bac_Node_01 ,3 ,AI ,0 100
AI_04 ,DA_Node_01 ,3 ,Server ,Bac_Node_01 ,4 ,AI ,0 100
Ao_01 ,DA_Node_01 ,4 ,Server ,Bac_Node_01 ,5 ,Av ,0 100
```

---

5 Sample Configuration File
```c
// Client Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name,Scan_Timeout,Data_Array_Name,Data_Array_Offset,Function,Node_Name,Length
ReadNode1,0s,DA_Node_01,0,RdA,Node_01,14
ReadNode2,0s,DA_Node_02,0,RdA,Node_01,14

// Server Side Nodes

Node_Name,Node_ID,Protocol
Bac_Node_01,01,BACnet_MSTP
Bac_Node_02,02,BACnet_MSTP

// Server Side Connections

Port,Baud,Protocol,Timeout,Connection_Type,Max_Master
R2,38400,BACnet_MSTP,30,MSTP_Master_Mode,127

// Client Side Connections

Con,Function,Node_Name,Node_ID
ReadNode1,0s,DA_Node_01,0
ReadNode2,0s,DA_Node_02,0

Of the 14 points stored they are stored at the following relative offsets in the Data Array

A's: 0,1,2,3
AO's: 4,5,6,7
BI's: Offset 8 (use 4 bits)
BO's: Offset 8 (use 4 bits)
Counters: Offset 10,11,12,13

```
Configuring the FieldServer as a Barrington MicroStar Server

This driver cannot be used to emulate an Barrington MicroStar Panel
Appendix 1 – Advanced Topics

Appendix 1.1. Scaling

Points CANNOT be scaled when they are read because each read returns 14 points of different types. They can be scaled by a task that executes on data update or a fixed frequency or they can be scaled when they are served.

Example: Scale before served. Ie scaled number is served.

Map_Descriptors

Map_Descriptor_Name,Data_Array_Name,Data_Array_Offset,Function,Node_Name,Node_Id,Data_Type,Data_Array_Low_Scale,Data_Array_High_Scale,Node_Low_Scale,Node_High_Scale

AI_01,DA_Node_01,0,Server,Bac_Node_01,1,100,0,100

AI_02,DA_Node_01,1,Server,Bac_Node_01,2,100,0,100
Appendix 1.2. Supported Communications functions

Always check the Data Sheet for an accurate and up to date list.

<table>
<thead>
<tr>
<th>Function</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poll Request with Download Data</td>
<td>Writes 4xAO values and 4xDO states. Can also be used to reset one/all of the 4 Counters.</td>
</tr>
<tr>
<td>Poll Request</td>
<td>Reads all data for all inputs. Analog inputs are 16 bits values.</td>
</tr>
</tbody>
</table>

Driver cannot read/write data to virtual microSTAR’s.
## Appendix 1.3. Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Resp</th>
<th>Format</th>
<th>Driver Ver.</th>
<th>Doc. Rev.</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008Feb11</td>
<td>PMC</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>Created</td>
</tr>
<tr>
<td>2014Sep21</td>
<td>PMC</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>Updated for new connection params</td>
</tr>
</tbody>
</table>