



Streaming Architecture for Control Networks Ethernet Driver FS-8705-102

Chipkin - Enabling Integration

salesgroup1@chipkin.com

Tel: +1 866 383 1657

© 2021 CHIPKIN AUTOMATION SYSTEMS

Driver Version: 0.0.1
Document Revision: 2

TABLE OF CONTENTS

1 SACN DESCRIPTION3

2 CONNECTION DESCRIPTION4

3 SACN CONFIGURATION5

 3.1 CREATE CONNECTION..... 5

 3.2 CREATE NODE..... 6

 3.3 CREATE TASK..... 7

 3.4 SAVING THE SERVER CONFIGURATION 8

 3.5 RESETTING THE SERVER CONFIGURATION 8

4 SACN TEST TOOLS9

 4.1 SACNVIEW..... 9

5 LICENSE10

 5.1 HOW TO GENERATE A PRE-KEY 10

 5.2 HOW TO ACTIVATE A PRODUCT KEY 11

6 FIRMWARE.....12

 6.1 UPLOADING THE SACN FIRMWARE 12

 6.2 HOW TO DOWNGRADE A SECURE QUICKSERVER TO INSECURE QUICKSERVER 12

7 IMPORTING AND EXPORTING CONFIGURATIONS.....13

 7.1 HOW TO EXPORT THE CONFIGURATION..... 13

 7.2 HOW TO IMPORT THE CONFIGURATION 13

 7.3 HOW TO IMPORT A PE CONFIGURATION..... 14

8 REVISION HISTORY15

APPENDIX A. TROUBLESHOOTING16

 APPENDIX A.1 DEBUGGING A SACN CONNECTION..... 16

 APPENDIX A.2 USING SACNVIEW FOR TESTING A SACN SERVER 16

 APPENDIX A.3 TESTING FIELDSEVER AS A SACN PASSIVE SERVER..... 20

APPENDIX B. EXAMPLE CONFIGURATION21

APPENDIX C. MARKETING24

 APPENDIX C.1 CASE STUDY 24

 APPENDIX C.2 KEYWORD..... 24

APPENDIX D. GLOSSARY OF TERMS25

1 SACN Description

The SACN Driver allows the FieldServer to transfer data from devices over Ethernet using the SACN protocol. The SACN Driver uses UDP. The default port is 5568 and is configurable.

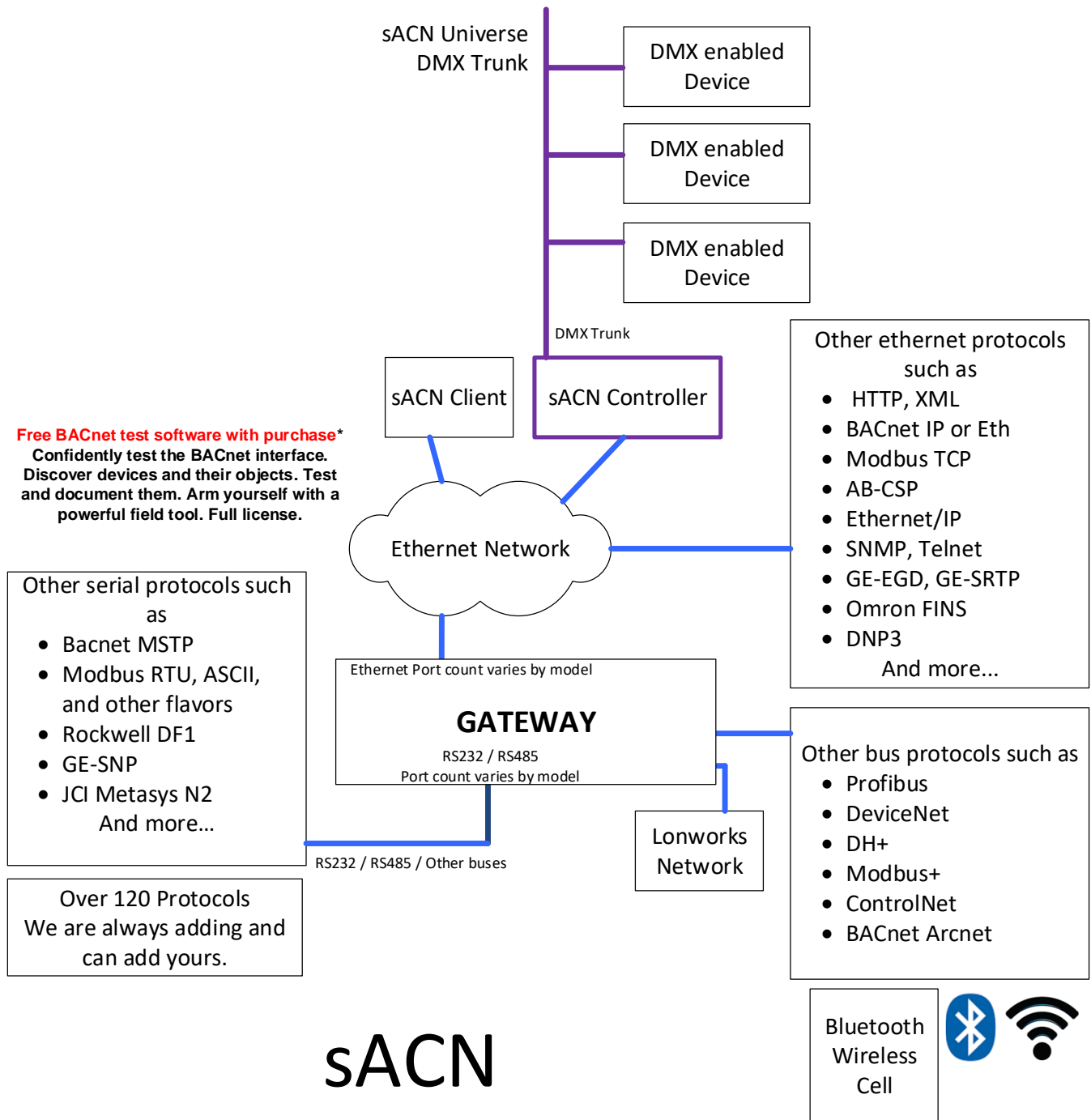
The driver was developed for the streaming Architecture for Control Networks (SACN) protocol. SACN is the common name for the ANSI E1.31-2016 standard and is used to transmit zero start code DMX data over a network in a way that is compatible with the full ACN standard, ANSI E1.17-2010.

The FieldServer emulates a passive Server, listening on the SACN network. When configured, the FieldServer listens for SACN messages containing DMX slot data. This data is stored on the FieldServer to be mapped to other protocols or simply to be viewed.

The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer.

2 Connection Description

This block diagram lists common network connections that can monitor SACN data using other protocols like Modbus® RTU/TCP, BACnet® and HTTP.



3 SACN Configuration

To configure the SACN driver, from the home page, visit the following link: http://{IP_ADDRESS}/sacnDriver/ui/

To configure the FieldServer, follow the instructions below to add a Connection (information on how to setup the passive server), Nodes (SACN universes), and finally Tasks (the data to extract from the DMX slot data).

3.1 Create Connection

To set up the FieldServer as a passive SACN server, first create a connection. The connection contains information about how to setup the server, like what port to listen to for SACN packets.

Connections

Name	Type	Parameters	Port	Actions
<div style="background-color: #007bff; color: white; padding: 5px 10px; display: inline-block; border-radius: 4px;">Create Connection</div>				

1. Click on the “Create Connection” button to open the Create Connection form.
2. Fill out the fields in the form. The fields are as follows:

COLUMN TITLE	FUNCTION	LEGAL VALUES
Name	Name of the server, used internally as an identifier	Text, must be unique
Type	The type of connection this is	ethernet
Parameters: Port	The physical port on the FieldServer to use	n1
Port	The UDP listening port for the SACN passive server	5568 , any available UDP port

* Bolded values are defaults

3. Click the “Save” button to add the connection.

If successful, the new entry will be populated in the Connections table:

Connections

Name	Type	Parameters	Port	Actions
sacnServer	Ethernet	{ "port": "n1" }	5568	Edit Delete

[Create Connection](#)

***Note*:** Only one server connection can exist. If multiple connections are created, only the first one will be used.

3.2 Create Node

Follow the instructions below universes for the SACN server to monitor.

Nodes

Name	Connection	Universe	Actions
Create Node			

1. Click on the "Create Node" button to open the Create Node form.
2. Fill out the fields in the form. The fields are as follows:

COLUMN TITLE	FUNCTION	LEGAL VALUES
Name	The name of the universe	Text, must be unique
Connection	The name of the server to add this universe.	Text (Use the name of the Connection created in the previous section)
Universe	The SACN universe to monitor	1-256

3. Click on the "Save" button to add the node.

If successful, the new entry will be populated in the Nodes table:

Nodes			
Name	Connection	Universe	Actions
Universe1	sacnServer	1	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

Repeat the above steps to add additional universes.

3.3 Create Task

Create tasks to add DMX slots to monitor

Tasks				
Name	Node	Data Broker	Slot Offset	Actions
<input type="button" value="Create Task"/>				

1. Click on the "Create Task" button to open the Create Task form.
2. Fill out the fields in the form. The fields are as follows:

COLUMN TITLE	FUNCTION	LEGAL VALUES
Name	The name of the variable to add.	Text, must be unique
Node	The node that this variable belongs to.	Text (Use the name of a node created in the previous section)
DataBroker: Name	The data array in the protocol engine to retrieve the value.	One of the Data Array names
DataBroker: Start	The starting offset in the array to retrieve the value	0 to ("Data_Array_length" - 1)
SlotOffset	The offset in the slot data that this data point represents	1-512

3. Click the “Save” button to add the task.

If successful, the new entry will be populated in the Tasks table:

Tasks					
Name	Node	Data Broker	Slot Offset	Actions	
OfficeCeilingLight1	Universe1	PE:DA_AI:1	1	Edit	Delete
OfficeCeilingLight2	Universe1	PE:DA_AI:2	2	Edit	Delete
CornerLight_R	Universe1	PE:DA_AI:3	3	Edit	Delete
CornerLight_G	Universe1	PE:DA_AI:4	4	Edit	Delete
CornerLight_B	Universe1	PE:DA_AI:5	5	Edit	Delete

[Create Task](#)

Repeat the above steps to add additional variables.

3.4 Saving the Server Configuration

When the configuration is complete, click on the “Save Configuration” button to save all of the updates and changes. For the configuration to take effect, reboot the system.



3.5 Resetting the Server Configuration

To clear the configuration and start over, click the “Reset Configuration” button. Then follow the instructions in the sections above to create new connections, nodes, and tasks.



4 SACN Test Tools

A list of SACN testing tools that you can use to test the functionality of your system.

4.1 sACNView

sACNView is a tool for viewing, monitoring and testing the ANSI/ESTA E1.17 protocol, informally known as “Streaming ACN”.

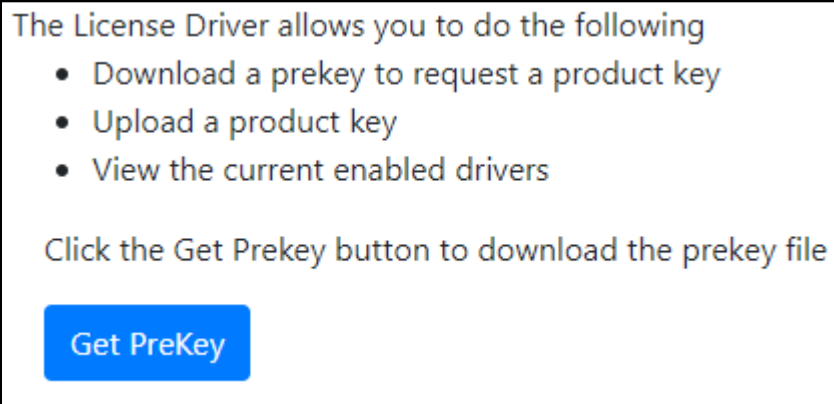
<https://sacnview.org/>

5 License

Some drivers such as SACN require a license product key. To generate this license product key a pre-key from the hardware is required.

5.1 How to generate a pre-key

1. Goto the license page http://{IP_ADDRESS}/chipkinLicenseDriver/ui/
2. Click the Get Pre-Key button.



A pre-key.txt file will be created and downloaded to your system. Send this pre-key.txt and your Job number (FSE1234) to Chipkin support.

5.2 How to activate a product key

Chipkin support can generate a license product key from the hardware pre-key. The product key will be sent as a text file via email.

1. Goto the license page http://{IP_ADDRESS}/chipkinLicenseDriver/ui/
2. Click "Browse" button and select the productkey-XXXXX.txt file provided to you by Chipkin Support.
3. Click the "Upload Product Key" button and wait for the product key to finish uploading.

Upload a product key. Select the product key to upload, then click the Upload Product Key button

The list of enabled product codes can be viewed in the "Enabled Product Codes" list.

Enabled Product Codes

The list of product codes that have been enabled by uploaded product keys

No product keys have been uploaded.

6 Firmware

6.1 Uploading the SACN Firmware

The SACN firmware does not come in the DCC000 default firmware from MSA-Safety. A custom firmware image needs to be loaded to enable the SACN driver.

The Quickserver must be in insecure mode before it will accept the SACN firmware. Follow the instructions in “How to downgrade a secure quickserver to insecure quickserver” section to downgrade the secure quickserver to insecure quickserver.

Once the quickserver is in insecure mode you can update the firmware through the FSGui using the SACN firmware.

6.2 How to downgrade a secure quickserver to insecure quickserver

The SACN driver firmware can not be loaded onto a secure quickserver, only an insecure quickserver. To downgrade a secure quickserver to insecure quickserver follow these instructions.

1. Download secure-to-insecure quickserver firmware image https://s.chipkin.com/files/uploads/2020/nov/secure-to-insecure_Chipkin-B0017-1.0.0-beta-armv7.simg
2. Use the Fieldserver toolbox to discover your Quickserver <https://www.sierramonitor.com/content/fieldserver-toolbox-0>
3. Browse to the FSGui firmware update page http://{IPADDRESS}/htm/fsgui.htm#20_OID
Setup => File Transfer, then click the Firmware tab on the right.
4. Click the “Choose File” button and select the secure-to-insecure quickserver firmware image “secure-to-insecure_Chipkin-B0017-1.0.0-beta-armv7.simg” then click the “Submit” button
5. Wait for the firmware to be uploaded. Then click the “System Reboot” button.
Note: System Reboot, not System Restart.

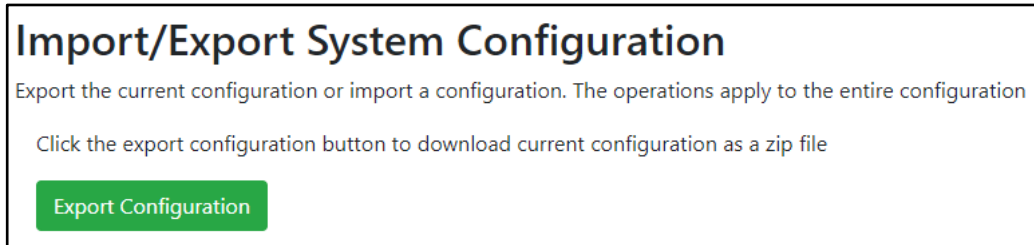
The Quickserver should now be in insecure mode.

7 Importing and Exporting Configurations

It is possible to export the current configuration to back it up or simply to make some edits. Users can also import either the entire configuration via a zip file or a PE (Protocol Engine) configuration.

7.1 How to Export the Configuration

1. Goto the system configuration page http://{IP_ADDRESS}/chipkinConfiguration/ui/
2. Click the Export Configuration button.



7.2 How to Import the Configuration

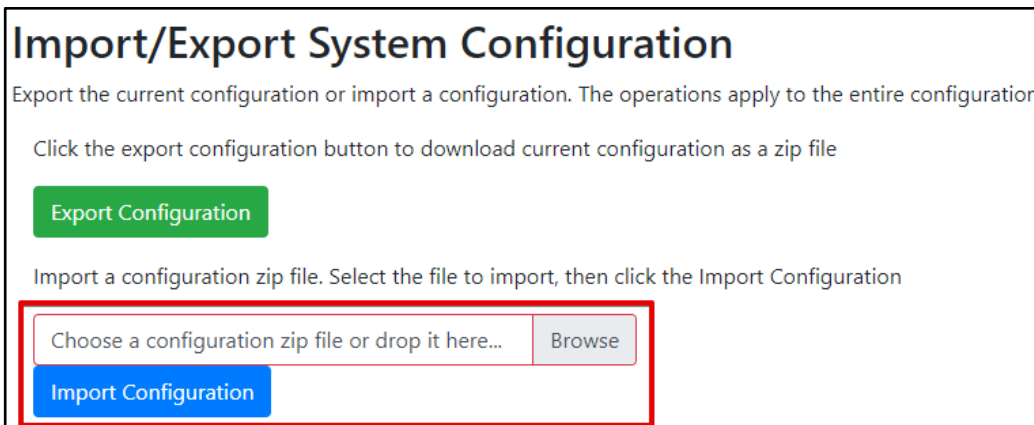
The file to import the configuration must be a zip file. The zip file should contain the following folders:

- ae - this folder contains any configuration files for the ae configuration
- documents - this folder contains any driver specific documents. For example, license product keys, etc.
- pe - this folder contains one config.csv file for the pe configuration.

To make sure the folder directory is correct, do an Export first, then extract the files, edit them, then zip them up again.

To import the configuration:

1. Goto the system configuration page http://{IP_ADDRESS}/chipkinConfiguration/ui/
2. Click the "Browse" button in the "Import/Export System Configuration" section and select the zip file containing the configuration to import.
3. Click the "Import Configuration" button and wait for the configuration to finish importing.
4. If successful, a success message will appear prompting a reboot of the Fieldserver for the changes to take effect.



7.3 How to Import a PE Configuration

It is possible to import a PE (Protocol Engine) configuration separately. To import a PE configuration:

1. Goto the system configuration page http://{IP_ADDRESS}/chipkinConfiguration/ui/
2. Click the “Browse” button in the “Import Specific Configuration” section and select the csv file containing the pe configuration to import.
3. Click the “Import PE Configuration” button and wait for the configuration to finish importing.
4. If successful, a success message will appear prompting a reboot of the Fieldserver for the changes to take effect.

Import Specific Configuration

Use the following control to import a specific portion of the configuration, this includes:

- Import PE Configuration

Import a pe configuration csv file. Select the file to import, then click the Import PE Configuration

8 Revision History

This table summarizes the update history for this document. Please contact Chipkin for an updated version of this document if required.

DATE	RESP	DOC. REV.	COMMENT
25 Nov 2020	ACF	1	Created initial document
2 June 2021	YC	2	Updated document format

Appendix A. Troubleshooting

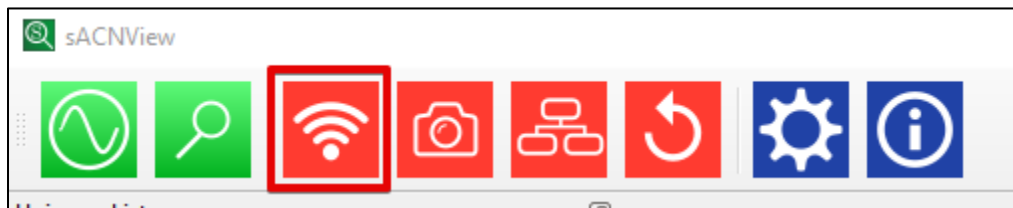
Appendix A.1 Debugging a SACN connection

- If the FieldServer is not receiving any SACN data, verify the IP Address and other network settings. SACN clients can send packets either via unicast or multicast.
- If sending by unicast, make sure that the network equipment that the FieldServer is connected to allows the traffic to be received by the FieldServer (i.e. a dumb hub, etc). Use wireshark on a computer connected to the same location as the FieldServer to see if there are any SACN packets. If there are not, that could mean the unicast messages are being routed directly to devices.
- If sending by multicast, make sure that the FieldServer's network settings are setup to be in the multicast range.

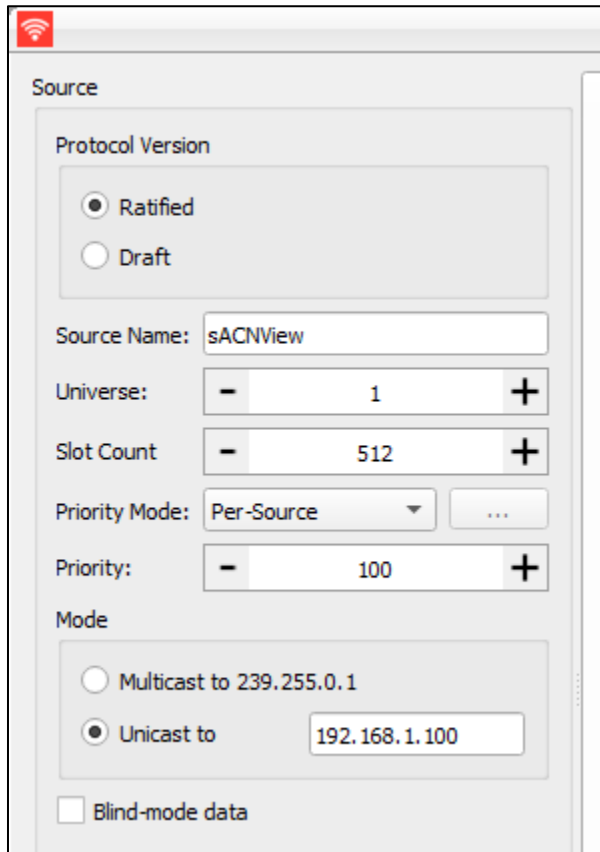
Appendix A.2 Using sACNView for Testing a SACN Server

Follow the steps in this section to setup the sACNView tool to test a connection to SACN Server (i.e. the FieldServer configured as a passive server).

1. Download and install the tool: <https://sacnview.org/>
2. In the tool, click on the Transmit button:



3. In the Transmit dialog box, Set the settings as follows:
 - Protocol Version: Ratified
 - Source Name: sACNView
 - Universe: 1 (or another configured universe)
 - Slot Count: 512
 - Priority Mode: Per-Source
 - Priority: 100
 - Mode: Unicast and set the IP Address of the FieldServer or SACN server



The screenshot shows a configuration window for the SACN Ethernet Driver. The window has a red Wi-Fi icon in the top-left corner. The main title is "Source".

Protocol Version

- Ratified
- Draft

Source Name:

Universe: (with minus and plus buttons)

Slot Count: (with minus and plus buttons)

Priority Mode: (with a dropdown arrow and an ellipsis button)

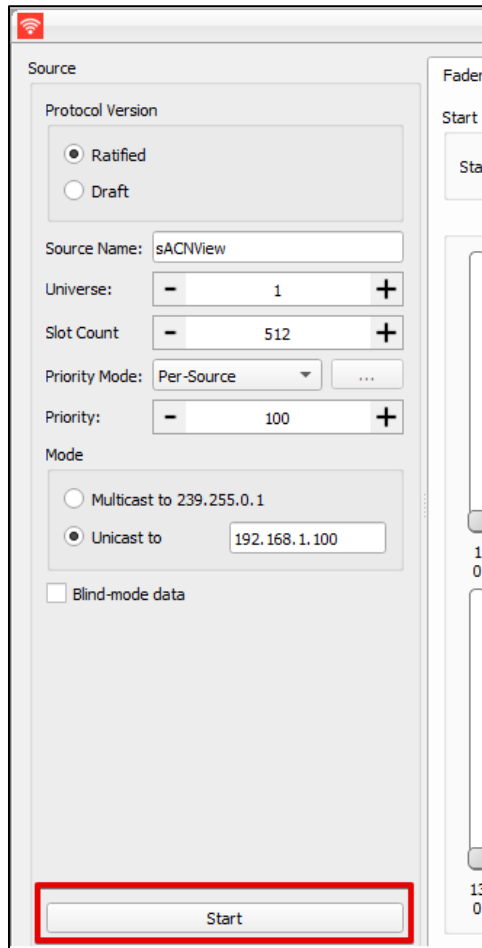
Priority: (with minus and plus buttons)

Mode

- Multicast to 239.255.0.1
- Unicast to

Blind-mode data

4. Click the Start button to begin transmitting data values



5. In the Faders Tab, change the sliding bar of slot 1 (or any other slot that has been configured) to change the value from 0 - 255

Transmit - Universe 1

Faders | Channel Check | Fade Range | Grid Control

Start at: - 1 +

Source

Protocol Version
 Ratified
 Draft

Source Name: sACNView

Universe: - 1 +

Slot Count: - 512 +

Priority Mode: Per-Source ...

Priority: - 100 +

Mode
 Multicast to 239.255.0.1
 Unicast to 192.168.1.100

Blind-mode data

Stop

1	2	3	4	5	6	7	8	9	10	11	12
67	0	0	0	0	0	0	0	0	0	0	0
13	14	15	16	17	18	19	20	21	22	23	24
0	0	0	0	0	0	0	0	0	0	0	0

Appendix A.3 Testing Fieldserver as a SACN Passive Server

The following instructions are how to confirm that a FieldServer that has been configured as a SACN Passive Server is working correctly.

1. Follow the instructions in Appendix A.2 to use sACNView to connect to the FieldServer configured as a SACN Passive Server.
2. Access the Data Array page on the FieldServer interface

The screenshot shows the sACNView interface. On the left is a 'Navigation' sidebar with a tree view. The 'View' folder is expanded, and 'Data Arrays' is selected. Under 'Data Arrays', 'DA_AI' is highlighted with a blue box, and the entire 'View' section is enclosed in a red rectangle. The main area displays the 'DA_AI' configuration. A 'Data Array' tab is active. Below it is a 'Data Array Attrib' table with the following data:

Data Array Attrib	
Name	
Data Array Name	DA
Data Format	Flo
Length in Items	20
Bytes per Item	4
Data Age	0.1

Below the attributes is a 'Display Format' dropdown menu set to 'Float'. At the bottom is a 'Data Array' table with the following data:

Data Array					
Offset	0	1	2	3	4
0	0.000000	0.000000	0.000000	0.000000	0.000000
10	10.000000	11.000000	0.000000	0.000000	0.000000
20	20.000000	21.000000	0.000000	0.000000	0.000000
30	0.000000	0.000000	0.000000	0.000000	0.000000

3. Modify the values in the sACNView for the slots that are mapped in the FieldServer.
4. Verify that the values update in the correct Data Array offsets.

Appendix B. Example Configuration

```
{
  "ae": {
    "sacnDriver": {
      "connections": [
        {
          "name": "Ethernet",
          "type": "ethernet",
          "parameters": {
            "port": "n1"
          },
          "port": 5568
        }
      ],
      "nodes": [
        {
          "connection": "Ethernet",
          "name": "Universe1",
          "universe": 1
        },
        {
          "connection": "Ethernet",
          "name": "Universe2",
          "universe": 2
        }
      ],
      "tasks": [
        {
          "node": "Universe1",
          "name": "CeilingLightRoom1",
          "slotOffset": 1,
          "dataBroker": {
            "pe": {
              "Name": "DA_AI",
```

```
        "Start": "1"
    }
}
},
{
    "node": "Universe1",
    "name": "CeilingLightRoom2",
    "slotOffset": 2,
    "dataBroker": {
        "pe": {
            "Name": "DA_AI",
            "Start": "2"
        }
    }
},
{
    "node": "Universe2",
    "name": "HallwayR5",
    "slotOffset": 1,
    "dataBroker": {
        "pe": {
            "Name": "DA_AI",
            "Start": "11"
        }
    }
},
{
    "node": "Universe2",
    "name": "StairwellG2",
    "slotOffset": 2,
    "dataBroker": {
        "pe": {
            "Name": "DA_AI",
            "Start": "12"
```

```
}  
  }  
    }  
  ]  
  }  
  }  
}
```

Appendix C. Marketing

Appendix C.1 Case Study

A series of case studies for sACN can be found here

[ToDo] – Add Case Study or link to Case Studies...

Appendix C.2 Keyword

sACN, Streaming-ACN, Streaming Architecture for Control Networks, e.131, ANSI E1.31-2016, DMX, ANSI E1.17-2010

Appendix D. Glossary of Terms

- **sACN** – Streaming Architecture for Control Networks
- **ACN** – Architecture for Control Networks: https://en.wikipedia.org/wiki/Architecture_for_Control_Networks
- **DMX** – Digital Multiplex, the standard digital communication protocol that is used to remotely control intelligent lighting fixtures.
- **UDP** - User Datagram Protocol