Crestron Fusion® Software SSI Model

Reference Guide

Crestron Electronics, Inc.

Original Instructions
The U.S. English version of this document is the original instructions.
All other languages are a translation of the original instructions.

Contents

Introduction

Programming is at the core of any successful Crestron Fusion® software deployment. The Single System Interface (SSI) is a programming model that ensures Crestron Fusion has consistent data sets across all devices regardless of the integrator, programming language, or application.

This document describes the data that applications must provide in order to be SSI compliant and provides guidelines to programmers that are implementing Crestron Fusion support in their code.

Since the SSI attributes are a standardized set of data that allows Crestron customers to use built-in reports in Crestron Fusion, it is critical that all applications provide data in a consistent manner. This document focuses on the application provider and includes all relevant implementation notes and testing instructions.

The SSI model provides the following:

- Signal naming conventions
- Behavior logic
- Join number guidelines
- Programmed asset models
- Scalability
- Predictable outcomes

Reports

Providing generated reports is one of the primary functions of Crestron Fusion software and the SSI model. Crestron Fusion reports typically include a visual chart of a particular data set. Each report has different customization options, including visualization, aggregation type, assets to display, and meeting type (depending on the report type), which allows for numerous different combinations.

The following reports are included in Crestron Fusion software:

- Device Usage by Meeting
- Device Usage Totals
- Meeting by Room
- Booking and Occupancy
- Call Statistics
- Device Usage
- Display Usage
- Display Usage (Current)
- System Usage
- Error Alerts
- Room Occupancy

Report Attributes

This section describes the attributes (signals) that are required to supply reports in Crestron Fusion with the necessary data.

DeviceUsage

The DeviceUsage attribute is used to report device usage in a room. Each device in the system is monitored for usage in minutes, and the usage string is sent to Crestron Fusion following the DeviceUsage string format.

Behavior: A device is considered in use when it is the current active source in the room and is presenting data.

DeviceUsage Attribute

Field	Value	Notes
Attribute Name	DeviceUsage	
Join	5	
Type	Serial	
Report Level	Room	
Expected Values	Formatted string as shown in the following sections	
Used in Reports	• Device Usage by Meeting • Call Statistics • Device Usage • Device Usage Totals • Display Usage	
C# Field	`FusionRoom.DeviceUsage.InputSig.StringValue`	

Device Usage Reporting

The data for Device Usage reporting is formatted as follows:

```
USAGE||Date_Stamp||Time_Stamp||TIME||Asset_Type||Asset_Name||-||Asset_
Usage||-||-||-||
```

Device Usage Reporting

Parameter	Meaning
Date_Stamp	Current date in YYYY-MM-DD format
Time_Stamp	Current time in HH:MM:SS format
Asset_Type	This field must match the Asset Type as registered in Crestron Fusion.

Parameter	Meaning
Asset_Name	This field must match the Asset Name as registered in Crestron Fusion.
Asset_Usage	The usage in minutes. If the value is less than zero, it is ignored.

The strings `USAGE` and `TIME` are fixed and must be in all caps.

Example: `USAGE||2016-10-26||12:34:39||TIME||Source||Room PC||-||4||-||-||-||`

The `USAGE` string must also be connected to the "Device Usage" signal pair Fusion room slot 3.

Display Usage Reporting

The data for Display Usage reporting is formatted as follows:

`USAGE||Date_Stamp||Time_Stamp||TIME||DISPLAY ||Asset_Name||-||Asset_Usage||-||-||-||`

Display Usage Reporting

Parameter	Meaning
Date_Stamp	Current date in YYYY-MM-DD format
Time_Stamp	Current time in HH:MM:SS format
Asset_Name	This field must match the Asset Name as registered in Crestron Fusion.
Asset_Usage	The usage in minutes. If the value is less than zero, it is ignored.

The strings `USAGE`, `DISPLAY`, and `TIME` are fixed and must be in all caps.

Example: `USAGE||2016-10-26||12:34:39||TIME||DISPLAY||Left Display||-||4||-||-||-||`

The application should not send Display Power Feedback, as Crestron Fusion will also generate these sessions.

Call Usage Reporting

The data for Call Usage reporting is formatted as follows:

`USAGE||Date_Stamp||Time_Stamp||CALL||Asset_Type||Dial_Type||Number||Call_Duration||Call_Status||-||-||`

Call Usage Reporting

Parameter	Meaning
Date_Stamp	Current date in YYYY-MM-DD format
Time_Stamp	Current time in HH:MM:SS format

Parameter	Meaning
Asset_Type	This field must match the Asset Type as registered in Crestron Fusion. This will be "Dialer" for Crestron Mercury® system, or it can be an "Audio Codec" or "Video Codec" as appropriate.
Dial_Type	For Crestron Mercury, this can be USB, SIP, Skype® conferencing solution, or Bluetooth® technology. For Video and Audio Codecs, the dial type will be "Video" or "Audio".
Asset_Usage	The usage in minutes. If the value is less than zero, it is ignored.
Number	The number (or address) dialed, "-" if not defined. For Skype, this "Call_Number" is variable depending on the connection (Meeting = URL, P2P = email address or number).
Call_Duration	Length of the call in minutes. If this field is less than one minute, the system rounds up.
Call_Status	Valid values are "success" or "failure". A "failure" value can be caused by a missed call, no answer, or no connection.

RoomOccupied

The RoomOccupied attribute is used to report occupancy in a room.

Behavior: A room is considered occupied when either of the following are true:

- An occupancy sensor in the room detects occupancy
- The room is in a call (drives occupancy high)

Use the FusionOccupancySensor class in SIMPL Sharp Pro or the Fusion Occupancy Sensor in SIMPL programming software to ensure that the proper signal is used.

NOTE: The Room_Occupied signal programming is required for Crestron Fusion to produce the corresponding occupancy sessions in reports. Room_Occupied must have logging enabled in Crestron Fusion.

RoomOccupied Attribute

Field	Value	Notes
Attribute Name	RoomOccupied	
Join	10	
Type	Asset (Occupancy Type)	
Report Level	Digital	
Expected Values	1 or 0	
Used in Reports	• Device Usage by Meeting • Meetings by Room	

Field	Value	Notes
	• Device Usage Totals • Room Occupancy • Booking and Occupancy	

DisplayPowerIsOn

Do not implement this attribute, as it conflicts with DeviceUsage and creates overlapping usage sessions in Crestron Fusion. Use the Fusion SSI Display Usage module instead.

DisplayUsage

The DisplayUsage attribute used to report display usage (projector and/or LCDs and other displays) in a room. This report requires that the DisplayUsage signal must be programmed for each room.

Display usage values in rooms with multiple projectors or multiple lamps/projectors should show the highest value in hours across all displays in the room. If the room has both projectors and LCDs/other displays, only lamps/projectors should be considered for this value. Individual displays report their usage to the Utilization attribute at the asset level and do not appear on the report.

NOTE: .AV Framework™ software currently sends this attribute for each individual asset.

DisplayUsage Attribute

Field	Value	Notes
Attribute Name	DisplayUsage	
Join	2	
Type	Analog	
Report Level	Room level	
Expected Values	Reported lamp hours	
Used in Reports	Display Usage (Current)	

OnlineStatus

The OnlineStatus attribute is determined by Crestron Fusion. It shows the status of the CIP connection between Crestron Fusion and the device.

ErrorMessage

The ErrorMessage attribute reports when a device is in an error state. Any device connected to the system must send a notification (a formatted error message) to Crestron Fusion when the device has a connection status change. The system detects the condition and sends the appropriate notification, including sending a "0": status on connection restored.

DisplayUsage Attribute

Field	Value	Notes
Attribute Name	ErrorMessage	
Join	2	
Type	Serial	
Expected Values	Formatted string as shown in the text following this table	
Used in Reports	Error Alerts	

The string must be formatted as follows: `<error severity>:<msg>` where `<error severity>` is a severity code ranging from 0 to 4, and `<msg>` is the error message.

The severity levels are as follows:

- 0 = OK (when the device is no longer in an error condition)
- 1 = Notice
- 2 = Warning
- 3 = Critical (A device that is not reporting but the room is still operational for meetings)
- 4 = Fatal (Any condition that renders the room or system unusable)

NOTE: All offline conditions will either be a 3 or 4 severity level.

Send an error notification whenever a device in the system stops reporting to the processor, and clear the error when it reconnects. Other notifications can be sent as determined by the programmer.

The message format for SSI is `<asset name>:<status>`.

Examples:

4:Display1 Offline

0:Display1 Online

3:DVDPlayer Offline

0:DVDPlayer Online

4:Display1 Offline

0:Display1 Online

3:Cable Box Offline

0:Cable Box Online

When the system first boots up, a timer should be created. If devices do not connect before the timer elapses, an error should be generated. The default time is 60 minutes.

SystemPowerIsOn

The SystemPowerIsOn attribute indicates that the room is in any state other than "Room Off", regardless of the device that is being used. This attribute is used to create the SystemUsage report. Power alone is not a consideration for this attribute; only system or room usage is considered.

SystemPowerIsOn Attribute

Field	Value	Notes
Attribute Name	SystemPowerIsOn	
Join	3	
Type	Digital	
Report Level	Room	
Expected Values	1 or 0	
Used in Reports	System Usage	

Approved Asset Types

The following table lists the approved Asset Types that should be used for consistency.

Asset Types

Asset Type	Notes	Severity
Access Point		4
Amplifier		3
AudioCodec		4
Blu-ray Disc® Player		3
Cable Box		3
CDPlayer		3
Computer		3
Controller		4
Dialer		4
Display		4
DM Receiver		3
DM Transmitter		3
DocumentCamera		3
DSP		4
DVDPlayer		3
Energy Load	Crestron Fusion® EM Asset	N/A
Energy Logging	Crestron Fusion® EM Asset	N/A
Energy Supply	Crestron Fusion® EM Asset	N/A
Keypad		4
Gateway		4
Lighting Load	Crestron Fusion® EM Asset	N/A
Lighting Scenes	Crestron Fusion® EM Asset	N/A
Media Player		3
Network Switch		4
Occupancy Sensor		3
Photocell		3
Power Conditioner		3
Presentation Gateway		4

Asset Type	Notes	Severity
Projector		4
Scaler		3
Schedule Display		3
Shade Load	Crestron Fusion® EM Asset	N/A
Shade Presets	Crestron Fusion® EM Asset	N/A
Switcher		4
Touchpanel	Includes preloaded drivers for all scheduling touch screens	4
UC Device		4
UC Software		3
VCR		3
VideoCapture	Includes preloaded driver for CaptureHD® system	3
VideoCodec	Includes preloaded driver for CaptureHD® system	4
Whiteboard		3

SSI Programming

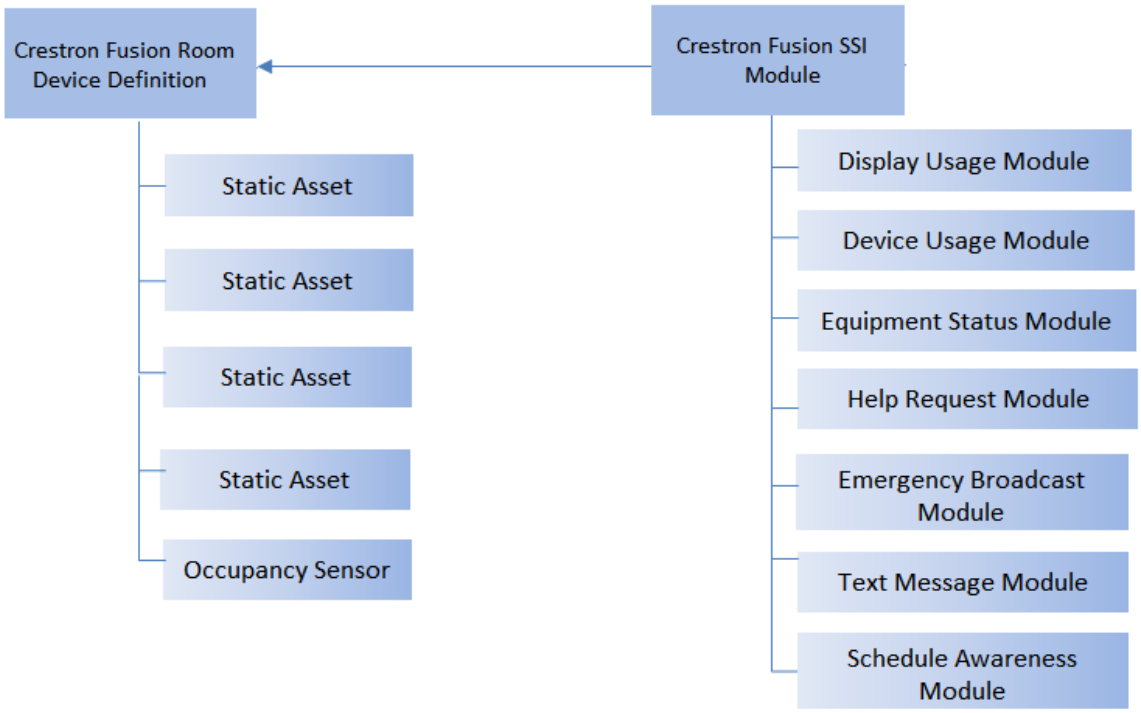
This section provides detailed information on programming SSI modules with SIMPL software for Crestron Fusion® software. The content in this section is intended for use by Crestron® software programmers. Use the provided procedures and signal explanations for each module to successfully program a Crestron Fusion deployment.

The following should be completed prior to programming with Crestron Fusion SSI modules:

- The scope of work has been determined and stakeholders have defined the functionality and features of the Crestron Fusion deployment.
- The programmer has completed the functional programming in the room. Functional programming is defined as any programming required that automates the room with control and feedback.
- A program has been created with a processor defined, and functional programming has been completed substantially.

The procedures in this document must be followed in order. Before using these procedures, review the following SIMPL programming flow diagram.

SIMPL Programming Flow Diagram



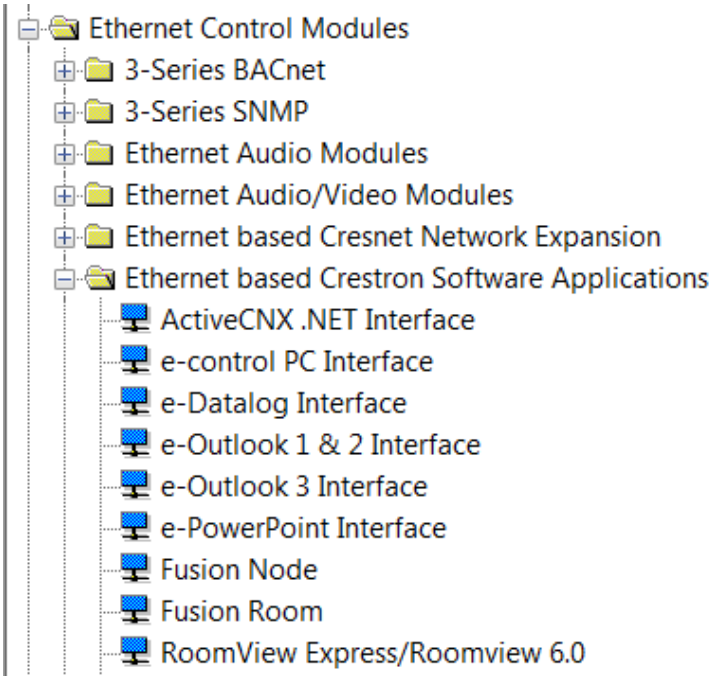
Add the Fusion Room Device Definition

The Crestron Fusion device definition (listed as "Fusion Room" in the Crestron Fusion database) is required for Crestron Fusion programming. The Fusion Room device definition permits information to pass to and from the Crestron Fusion server and is the base for implementing Crestron Fusion code to the program.

To add the Fusion Room device definition in SIMPL:

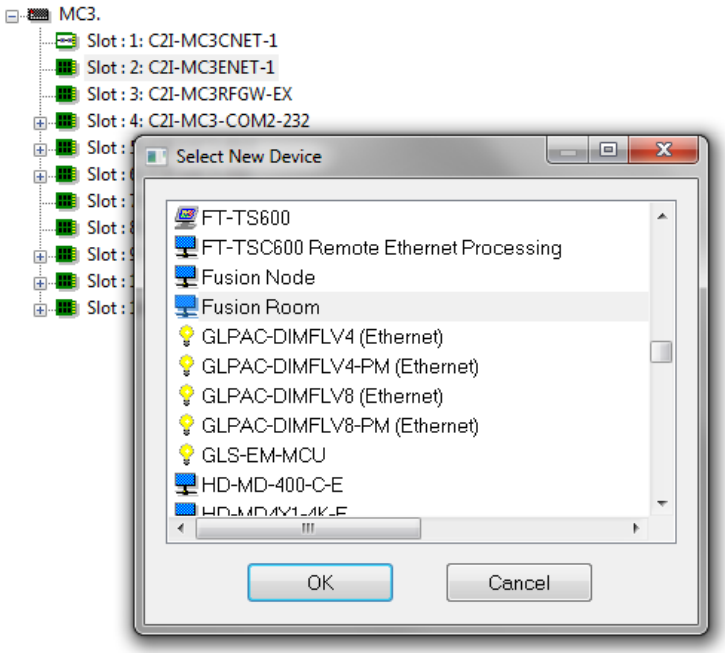
1. Ensure all prerequisites have been met as described above.
2. Add the **Fusion Room** device definition to the program. The definition is located under **Crestron Devices** > **Ethernet Control Modules** > **Ethernet based Crestron Software Applications**.

   Fusion Room Device Definition



The user can also right-click the **Ethernet** subslot for the processor and then add the **Fusion Room** device definition.

Ethernet Subslot



Add Assets to the Fusion Room Device Definition

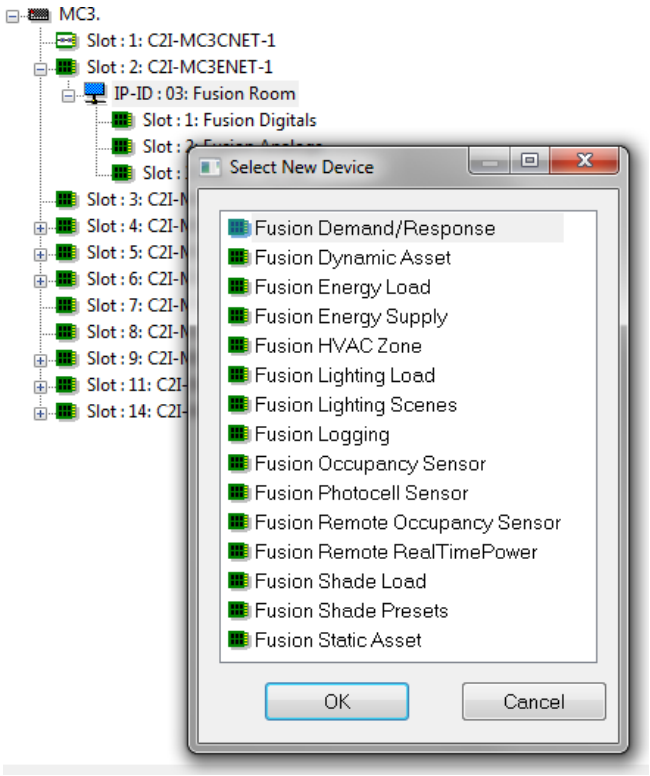
Assets in Crestron Fusion encompass a broad spectrum of items in a room, ranging from AV equipment to furniture, drapes, and lighting fixtures. The Static Asset is used to define these assets.

NOTE: The Fusion Room device definition must be added to the program prior to adding assets. For more information, refer to Add the Fusion Room Device Definition (on page 12).

To add assets to the Fusion Room device definition in SIMPL:

1. Ensure all prerequisites have been met as described above.
2. Select the **Configure** view.
3. Right-click the **Fusion Room** device definition.
4. Select **Add Item to Fusion Room**. The **Select New Device** dialog box is displayed.
5. Select **Fusion Static Asset** from the list.

Select New Device Dialog Box



6. Repeat steps 1–5 for all Assets that need to be added into Crestron Fusion. For an occupancy sensor, select **Fusion Occupancy Sensor** from the **Device** list.

   NOTE: Some examples of Assets are sources such as a DVD Player, AV Switcher, PC, AirMedia® presentation gateway device, and so forth. Any piece of equipment entered for tracking in Crestron Fusion should be added as a Static Asset in programming.

Add the Fusion SSI Module to a Program

The purpose of this section is to create a common programming practice for the additional modules required for Crestron Fusion.

NOTE: The Fusion Room device definition and any assets must be added to the program prior to implementing the SSI module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add Assets to the Fusion Room Device Definition (on page 14), respectively.

The Fusion SSI module is designed to attach to the Fusion Room device definition in SIMPL. Right-clicking the **Fusion SSI module** and dragging it to the **Fusion Room** device definition creates signals between the module and the definition. This pairing automatically generates the necessary signals to attach the module and the device together.

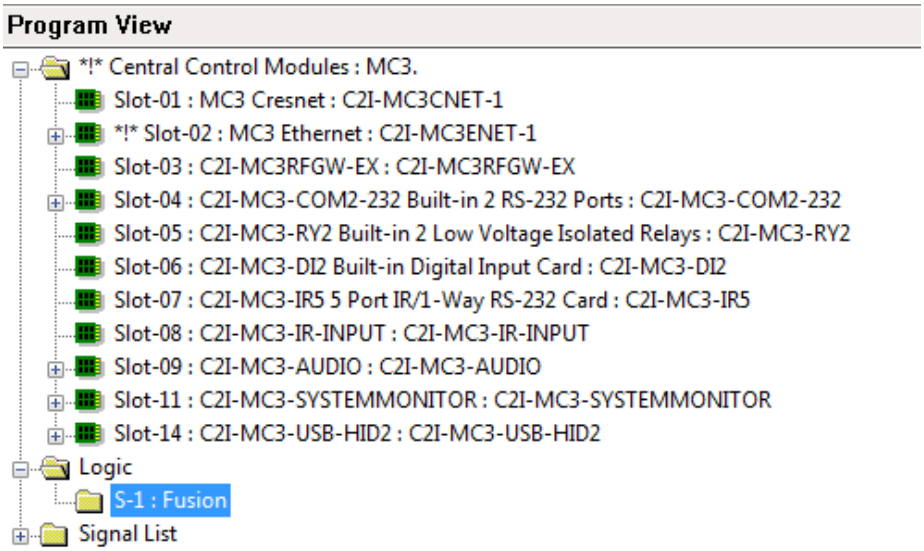
From the prompt that displays, assign a prefix for the signal names. The most common prefix name is the room's name (for example, Conference Room 101, Classroom 200, and so forth.)

Programming Procedure

To add the Fusion SSI Module to a Program:

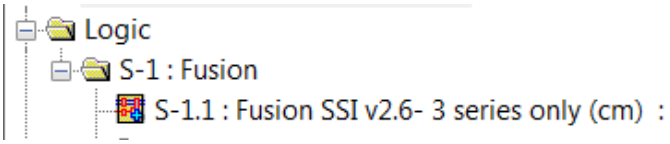
1. Ensure all prerequisites have been met as described above.
2. Create a subfolder named "Fusion" in SIMPL.

Program View



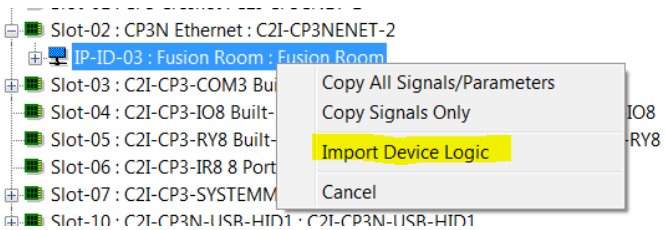
3. Add the Fusion SSI module to the Fusion subfolder. The module is located in the Fusion folder under Crestron Modules.

   Fusion SSI Module Added to Fusion Subfolder

4. Right-click the Fusion SSI module and drag it into the Fusion Room device definition. Release the button and select **Import Device Logic** from the context menu.

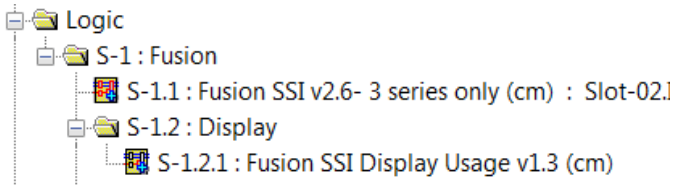
   Adding Fusion SSI Module to Fusion Room Device Definition

   NOTE: For the remaining steps in this procedure, the latest version of the indicated modules must always be used.

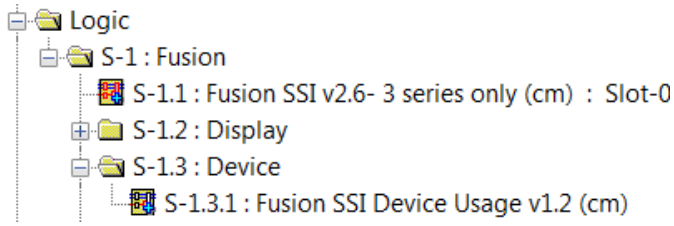
5. Create a subfolder under the Fusion subfolder named "Display", and then add the **Fusion SSI Display Usage** module to the Display subfolder.

   Fusion SSI Display Usage Module

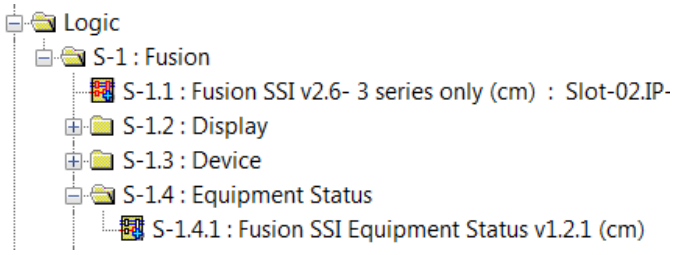
6. Create a subfolder under the Fusion subfolder named "Device", and then add the **Fusion Device Usage** module to the Display subfolder.

   Fusion SSI Device Usage Module

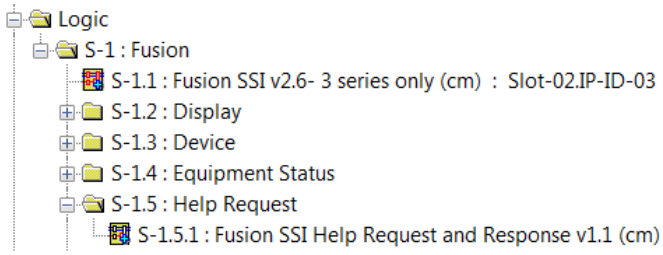
7. Create a subfolder under the Fusion subfolder named "Equipment Status", and then add the **Fusion SSI Equipment Status** module to the Equipment Status subfolder.

   Fusion SSI Equipment Status Module



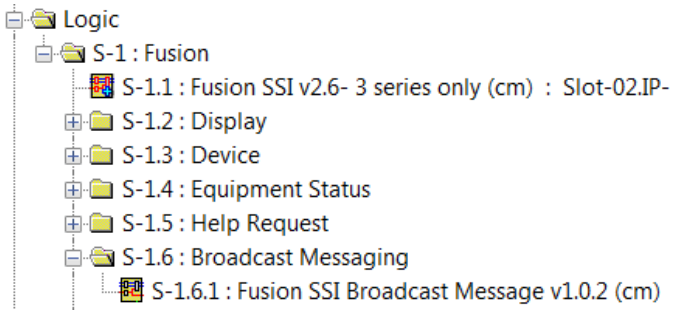
8. Create a subfolder under the Fusion subfolder named "Help Request" (for help messaging), and then add the **Fusion SSI Help Request and Response** module to the Help Request subfolder.

   Fusion SSI Help Request and Response Module



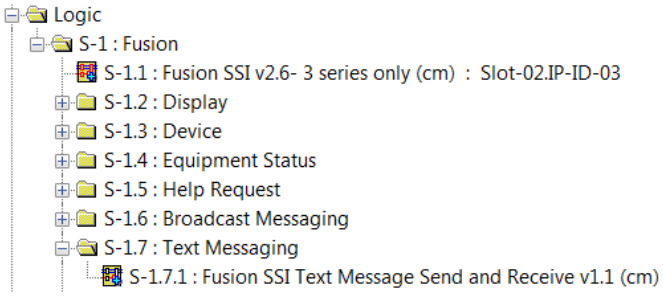
9. Create a subfolder under the Fusion subfolder named "Broadcast Messaging", and then add the **Fusion SSI Broadcast Message** module to the Broadcast Messaging subfolder.

   Fusion SSI Broadcast Message Module



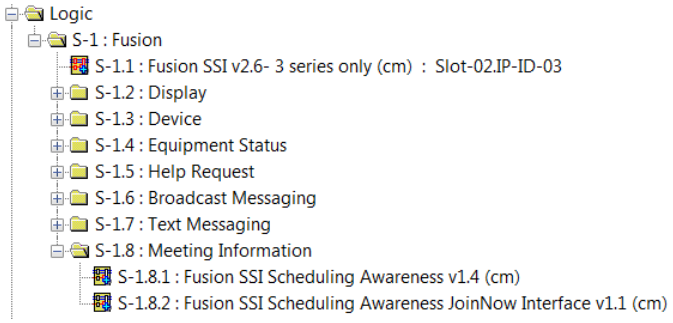
10. Create a subfolder under the Fusion subfolder named "Text Messaging", and then add the **Fusion SSI Text Message Send and Receive** module to the Text Messaging subfolder.

    Fusion SSI Text Message Send and Receive Module

11. Create a subfolder under the Fusion subfolder named "Meeting Information", and then add the **Fusion SSI Scheduling Awareness** module to the Meeting Information subfolder.

    Fusion SSI Scheduling Awareness Module

NOTE: There are additional modules available for Crestron Fusion programming. For more information on these modules, refer to the embedded SIMPL help files.

Signal Definitions

Refer to the following tables for Fusion SSI module signal definitions.

Fusion SSI Module Inputs

Signal Group	Module Signal Name	Definition
Fusion Room	{{Fusion_Digitals_>> _ SystemPowerOn}}	This signal links to the **SystemPowerOn** signal on the **Fusion Room** device definition and is used when System Power has been triggered to On from Crestron Fusion.
Fusion Room	{{Fusion_Digitals_>> _ SystemPowerOff}}	This signal links to the **SystemPowerOff** signal on the **Fusion Room** device definition and is used when System Power has been triggered to Off from Crestron Fusion.
Fusion Room	{{Fusion _Digitals_>> _ DisplayPowerOn}}	This signal links to the **DisplayPowerOn** signal on the **Fusion Room** device definition and is used when Display Power has been triggered to On from Crestron Fusion.
Fusion Room	{{Fusion_Digitals_>> _ DisplayPowerOff}}	This signal links to the **DisplayPowerOff** signal on the **Fusion Room** device definition and is used when Display Power has been triggered to Off from Crestron Fusion.
Fusion Room	{{Fusion _Digitals_>> _ AuthenticateSucceeded}}	This signal links to the **AuthenticateSucceeded** signal on the **Fusion Room** device definition, indicating that a user has been authenticated from Crestron Fusion.
Fusion Room	{{Fusion _Digitals_>> _ AuthenticateFailed}}	This signal links to the **AuthenticateFailed** signal on the **Fusion Room** device definition, indicating that a user authentication has failed in Crestron Fusion.
Fusion Room	{{Ethernet_Offline_Manager_>>_ offline}}	This signal links to an **Ethernet Offline Manager update-request** signal. The Ethernet Offline Manager must be added to the **Fusion Room** device definition manually.
Fusion Room	{{Fusion_Analogs_>> _ BroadcastMsgType}}	This signal links to the **BroadcastMsgType** output and receives an analog value for the type of broadcast message that was sent (0–9d).

Signal Group	Module Signal Name	Definition
Fusion Room	{Fusion_Serials_>>_ HelpResponse}}	This signal links to the **HelpResponse** output of the **Fusion Room** device definition and contains the response message sent from Crestron Fusion to the room.
Fusion Room	{{FusionRoom_Serials_>> _ TextMessageFromFusion}}	This signal links to the data sent from Crestron Fusion back to the Crestron Fusion SSI module.
Fusion Room	{{Fusion_Serials_>> _ BroadcastMsg}}	This signal links to the **BroadcastMsgType** output and receives an analog value for the type of broadcast message that was sent (0–9d).
Fusion Room	{{Fusion_Serials_>>_ GroupMembership}}	This signal links to the data sent from Crestron Fusion back to the Crestron Fusion SSI module.
Fusion Room	{{Fusion_RoomView_Scheduling_ Data_>>_ScheduleResponse$}}	This input receives scheduling data (in XML format) that can then be processed by the module.
Fusion Room	{{Fusion_RoomView_Scheduling_ Data_>>_CreateResponse$}}	This input issues an XML formatted request for scheduling data.
Fusion Room	{{Fusion_RoomView_Scheduling_ Data _>>_RemoveResponse$}}	This input issues an XML formatted clear command to remove all responses.
	{{Fusion_Room_Data_>> _ RoomAutomationResponse$}}	This signal links to the **RoomAutomationResponse$** signal on the **Fusion Room Data** subslot. The data that is sent through this signal contains automation rules in XML format and is stored on the processor.
	{{Fusion_Room_Data_>> _ RoomTimeclockResponse$}}	This signal links to the **RoomTimeclockResponse$** signal on the **Fusion Room Data** subslot. Data that is sent through this signal contains time sync information in XML format that is then parsed and used to update the processor date/time. This signal is used by the **Fusion Time Sync** module.
	{{Fusion_Room_Data_>> _ RoomConfigResponse$}}	This signal links to the **RoomConfigResponse$** signal on the **Fusion Room Data** subslot. The data sent through this signal contains room details in XML format. These could be custom properties, settings, and so forth.

Signal Group	Module Signal Name	Definition
	{{Fusion_Room_Data_>> _ RoomListResponse$}}	This signal links to the **RoomListResponse$** signal on the **Fusion Room Data** subslot. Data that is sent through this signal contains information for room availability. This data is sent in XML format.
	{{Fusion_Room_Data_>> _ RoomAttributeResponse$}}	This signal links to the **RoomAttributeResponse$** signal on the **Fusion Room Data** subslot. Data sent through this signal contains information about attributes from Crestron Fusion. This data is sent in XML format.
	{{Fusion_Room_Data_>> _ RoomAssetQueryResponse$}}	This signal links to the **RoomAssetQueryResponse$** signal on the **Fusion Room Data** subslot. Data contained in this signal lists Assets associated to the room. This data is sent in XML format.
	{{Fusion_Room_Data_>> _ AssetDriverQueryResponse$}}	This signal links to the **RoomDriverQueryResponse$** signal on the **Fusion Room Data** subslot. This data contains information related to a dynamic driver built for an Asset. The response can contain information values about the dynamic driver. The data is in XML format.
	{{Fusion_Room_Data_>> _ ActionQueryResponse$}}	This signal links to the **RoomActionQueryResponse$** signal on the **Fusion Room Data** subslot. Data contained in this signal is response information to push registration requests, Meeting ID requests, and meeting end time change requests. The data is in XML format.
	{{Fusion_Room_Data_>> _ LocalDateTimeQueryResponse$}}	This signal links to the **LocalDateTimeQueryResponse$** signal on the **Fusion Room Data** subslot. The data contained in this signal returns the current date and time for the room. The data is formatted in XML.

Signal Group	Module Signal Name	Definition
	Update_System_Info	This signal links to the logic and lets Crestron Fusion know that the system and program has started. This signal is important to use because of the GUID Updater built into the Crestron Fusion SSI Module.
	System_Is_On	This digital feedback signal links to the programming logic for overall system status.
	[Display_Is_On]	This digital feedback signal links to the display's control module and provides current power status for the primary display.
	[Lamp_Hours]	This analog signal links to the display's control module and provides current total lamp hours reporting from the projector.
	Device_Usage_String$	This signal links to the Device Usage module and provides data to send to Crestron Fusion.
	Equipment_Status$	This signal links to the Crestron Fusion SSI Equipment Status module and provides data to send to Crestron Fusion.
	[Log_Text$]	This signal links to the **LogText** signal on the Fusion Room device definition and sends custom serial data strings to Crestron Fusion which can be used for logging nonstandard information to the database.
	[Help_Request$]	This signal links to the **Help_Request$** serial output signal on the Crestron Fusion SSI Help Request and Response module.
	[Text_Message_TX$]	This signal links to the **Text_Message_TX$** serial output signal on the Crestron Fusion SSI Text Message module.
Time_Sync	Sync_Time_b	When Use_Time_Sync is set to "Yes", this digital signal (when triggered) sends a serial string out the **Local_Date_Time_Query_tx$** signal to Crestron Fusion to retrieve the current time and date information. This signal is usually tied to a When module.

Fusion SSI Module Outputs

Signal Group	Module Signal Name	Definition
Fusion Room	{{SystemPowerIsOn_>>_ Fusion _Digitals}}	This is the digital signal value for **System Power On**. This signal provides feedback from the Crestron Fusion SSI module to Crestron Fusion.
Fusion Room	{{DisplayPowerIsOn_>>_ Fusion _Digitals}}	This is the digital signal value for **Display Power On**. This signal provides feedback from Crestron Fusion SSI module to the Crestron Fusion.
Fusion Room	{{MsgBroadcastEnabled_ >>_ Fusion_Digitals}}	This signal indicates whether broadcast messaging can be invoked for this room.
Fusion Room	{{update-request_>> _ Ethernet_Offline_ Manager}}	This digital signal is tied to the Ethernet Offline Manager that sends a trigger to request an update for network online or offline status.
Fusion Room	{{DisplayUsage_>>_Fusion_ Analogs}}	This analog signal ties to the **Display Usage** analog input for sending current lamp hours to Crestron Fusion.
Fusion Room	{{HelpRequest_>>_Fusion _ Serials}}	This serial signal sends Help Requests to Crestron Fusion. This signal is passed through the Crestron Fusion SSI module from the Help Request module.
Fusion Room	{{ErrorMessage_>>_Fusion _Serials}}	This serial signal sends Equipment Status messages to Crestron Fusion. This signal is passed through the Crestron Fusion SSI module from the Equipment Status module.
Fusion Room	{{LogText_>>_Fusion_ Serials}}	This serial output contains custom serial data which can be used for logging information to the Crestron Fusion database.
Fusion Room	{{DeviceUsage_>>_Fusion _ Serials}}	This serial signal sends Device Usage data to Crestron Fusion. This signal is passed through the Crestron Fusion SSI module from the Device Usage module.
Fusion Room	{{TextMessage_>>_Fusion _ Serials}}	This serial signal sends Text Messages to Crestron Fusion. This signal is passed through the Crestron Fusion SSI module from the Text Message module.
Fusion Room	{{BroadcastMsgResponse_ >> _Fusion_Serials}}	This serial signal sends an Emergency Response message to Crestron Fusion. This signal is passed through the Crestron Fusion SSI module from the Emergency Broadcast module.

Signal Group	Module Signal Name	Definition
Fusion Room	{{FreeBusyStatus_>>_ Fusion _Serials}}	This serial string contains scheduling data to determine whether a room is available or not available.
Fusion Room	{{AuthenticateRequest_>> _Fusion_Serials}}	This serial string is used to authenticate users against a database. This string can then be used for accessing the room user interfaces or other security.
Fusion Room	{{Fusion_ScheduleQuery$_ >> _RoomView_Scheduling_ Data}}	This serial signal returns XML formatted query information for scheduling associated to the room.
Fusion Room	{{Crestron_ CreateMeeting$_>> _ RoomView_Scheduling_ Data}}	This serial signal returns XML formatted data for scheduling associated to the room.
Fusion Room	{{Fusion_ RemoveMeeting$_>> _ RoomView_Scheduling_ Data}}	This serial signal returns XML formatted data to remove scheduling associated to the room.
	{{RoomAutomationQuery$_ >> _Fusion_Room_Data}}	This signal links to the **RoomAutomationQuery$** signal on the **Fusion Room Data** subslot. The data sent through this signal contains request information for room automation rules. The data is in XML format.
	{{RoomTimeclockQuery$_ >> _Fusion_Room_Data}}	This signal links to the **RoomTimeclockQuery$** signal on the **Fusion Room Data** subslot. The data sent through this signal contains the request for time sync information in XML. This signal is used by the Fusion Time Sync module.
	{{RoomConfigQuery$_>> _ Fusion_Room_Data}}	This signal links to the **RoomConfigQuery$** signal on the **Fusion Room Data** subslot. The data sent through this signal contains the request for room details in XML format, for example, custom properties, settings, etc.
	{{RoomListQuery$_>>_ Fusion _Room_Data}}	This signal links to the **RoomListQuery$** signal on the **Fusion Room Data** subslot. The data sent through this signal contains the request for information on room availability. This data is sent in XML format.
	{{RoomAttributeQuery$_>> _Fusion_Room_Data}}	This signal links to the **RoomAttributeQuery$** signal on the **Fusion Room Data** subslot. The data sent through this signal contains request information about attributes from Crestron Fusion. The data is sent in XML format.

Signal Group	Module Signal Name	Definition
	{{RoomAssetQuery$_>>_ Fusion_Room_Data}}	This signal links to the **RoomAssetQuery$** signal on the **Fusion Room Data** subslot. The data contained in this signal is requested for Assets associated to the room. The information is sent in XML format.
	{{AssetDriverQuery$_>> _ Fusion_Room_Data}}	This signal links to the **AssetDriverQuery$** signal on the **Fusion Room Data** subslot. The data contains request information related to a dynamic driver built for an Asset.
	{{ActionQuery$_>>_Fusion _Room_Data}}	This signal links to the **ActionQuery$** signal on the **Fusion Room Data** subslot. The data contained in this signal is a request to push register, request Meeting ID, and request a meeting end time change. The data is in XML format.
	{{LocalDateTimeQuery$_>> _CrestronFusion_Room_ Data}}	This signal links to the LocalDateTimeQuery$ signal on the **Fusion Room Data** subslot. The data contained in this signal requests the current date and time for the room. The data is in XML format.
System Information	[Processor_DHCP_ON]	This serial string provides feedback indicating if DHCP is On for the processor.
System Information	[Processor_Model]	This serial string provides feedback for the model of the processor.
System Information	[Processor_Firmware]	This serial string provides feedback on the current firmware running on the processor.
System Information	[Firmware_Date]	This serial string provides the release date for the current running firmware version.
System Information	[Processor_MAC]	This serial string provides the MAC address of the processor.
System Information	[Processor_IP]	This serial string provides the current IP address of the processor.
System Information	[Processor_Hostname]	This serial string provides the current hostname of the processor.
System Information	[Processor_Serial]	This serial string provides the serial number of the processor.
System Information	[Processor_Uptime]	This serial string provides the time since the last time the processor rebooted. **Do not send this value to Crestron Fusion more than once every 10 minutes.**

Signal Group	Module Signal Name	Definition
System Information	[Program_Uptime]	This serial string provides the time since last restart or upload of the program running in Slot 1. **Do not send this value to Crestron Fusion more than once every 10 minutes.**
System Information	[Programmer_Name]	This serial string provides the name of the programmer for the current running program, if available.
System Information	[System_Name]	This serial string provides the name of the system for the current running program, if available.
System Information	[Source_File]	This serial string provides the location path to the current running program, for example, c:\Program Files\Crestron\program_name.lpz).
System Information	[Compile_Date]	This serial string provides the date that the current program was compiled on.
	Fusion_System_On_Request	This digital signal passes the signal value from Crestron Fusion through the Crestron Fusion SSI module and then sends the value to programming within the room for System Power On.
	Fusion_System_Off_Request	This digital signal passes signal value from Crestron Fusion through the Crestron Fusion SSI module and then sends the value to programming within the room for System Power Off.
	[Fusion_Display_On_Request]	This digital signal passes the signal value from Crestron Fusion through the Crestron Fusion SSI module and then sends the value to programming within the room for Display Power On.
	[Fusion_Display_Off_Request]	This digital signal passes the signal value from Crestron Fusion through the Crestron Fusion SSI module and then sends the value to programming within the room for Display Power Off.
	[Help_Response$]	This pass-through string sends XML data from Crestron Fusion to the Help Request and Response module. This data is then formatted by the module and sent to a user interface.
	[Broadcast_Message_Type]	This pass-through string sends the analog value (0d–9d) to the Crestron Fusion SSI Broadcast Message module.

Signal Group	Module Signal Name	Definition
	[Broadcast_Message$]	This pass-through string sends the serial string broadcast message to the Crestron Fusion SSI Broadcast Message module.
	[Text_Message_RX$]	This pass-through string sends XML data from Crestron Fusion to the Crestron Fusion Text Message module. This data is then formatted by the module and sent to a user interface.
	[to_PrograminitComplete]	This signal is used to ensure that the GUID updater is finished processing before sending the RVI file to Crestron Fusion.
Fusion EM	[GUID_Update_Busy]	This signal indicates that the internal GUID Update module is currently running. When low, the process has completed. This signal can be used to trigger events in programming after the GUID Update has been completed.
Fusion EM	[RoomName$]	This signal returns the name of the Crestron Fusion room.
Fusion EM	[RoomGUID$]	This signal returns the GUID information created by the GUID Updater.
Fusion EM	[SymbolSlotGUIDs$]	This signal returns the GUID information with the symbol slot information. This is important if there are multiple symbols in a program. This information identifies which GUID is associated to which symbol.

Fusion SSI Module Parameters

Module Signal Name	Definition
Processor_Type	This parameter defines whether the system is a standard Crestron processor or a DMPS system.
Primary_SSI	This parameter is set for multiple Crestron Fusion SSI modules in one program. Designating one primary prevents the GUID Updater (within all other Crestron Fusion SSI modules) from generating updated .rvi files. One module per program should be set as the primary. All other modules should have **No** selected.
Prefix_GUID	This parameter defines if the GUID Updater prefix is invoked within the module. The prefix consists of the TSID and Slot number. It is recommended to always use this prefix to ensure unique GUIDs across all programs. SIMPL # programmers need to ensure that GUIDs are used in their programs.
Append_Program_Slot	This parameter determines if the Program Slot number is prefixed to the GUID at program start.

Module Signal Name	Definition
File_Location	This parameter defines the location for the .rvi file that Crestron Fusion reads to invoke symbol discovery.
Fusion_SSI_ID	Future use, but a value is required to complete the module. If multiple SSI modules are in a program, the value must be unique across modules.
Scheduling_ID	The value entered must be the same ID used for the Scheduling ID parameter on the Fusion SSI Scheduling Awareness module. If the IDs do not match, the Scheduling Awareness module will not request the message body contained in the scheduled events from Crestron Fusion.

Program the Time Functions

The Fusion SSI module can synchronize the time and date of both the processor and the Crestron Fusion server. This procedure identifies the Fusion SSI module settings necessary to issue a command to Crestron Fusion so that Crestron Fusion can send back the current time and date. The Fusion SSI module then updates the time and date on the processor.

NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to programming time functions. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

Programming Procedure

To program the time functions in the Fusion SSI module:

1. Ensure all prerequisites have been met as described above.
2. Set the **Use_Time_Sync** parameter to "Yes".

   NOTE: If using multiple Fusion SSI modules in a processor, even if it's running in different program slots, only one should be set to "Yes" so multiple versions are not setting the time on the same system.

3. Add a **When** module to the **Date/Time** folder to drive the **Sync_Time_b** input signal.

Program the Fusion SSI Display Usage Module

Crestron Fusion allows users to track usage information for displays, whether they are LCD, LED, projectors, or other supported types. The Fusion SSI Display Usage module generates and formats usage data (in minutes) that is sent to the database. The information is available in the Crestron Fusion reports section.

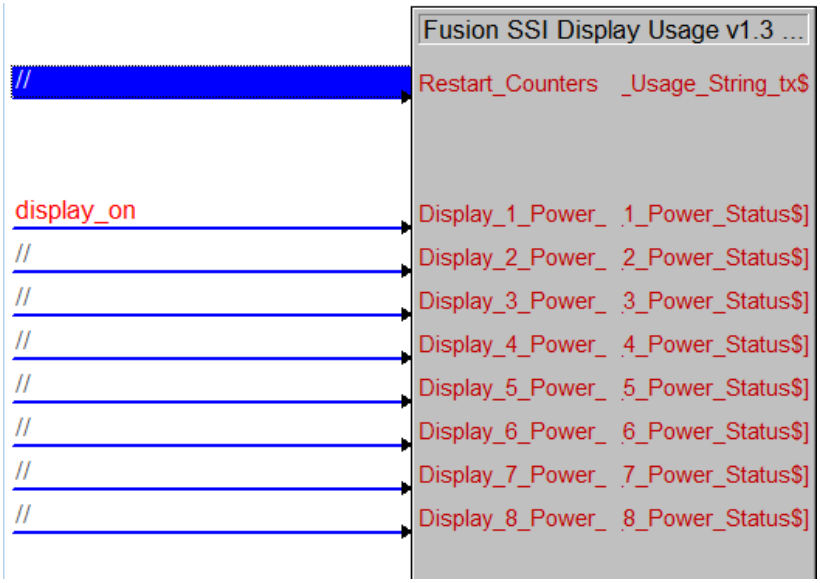
NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to implementing the Fusion SSI Display Usage module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

Programming Procedure

To program the Fusion SSI Display Usage module:

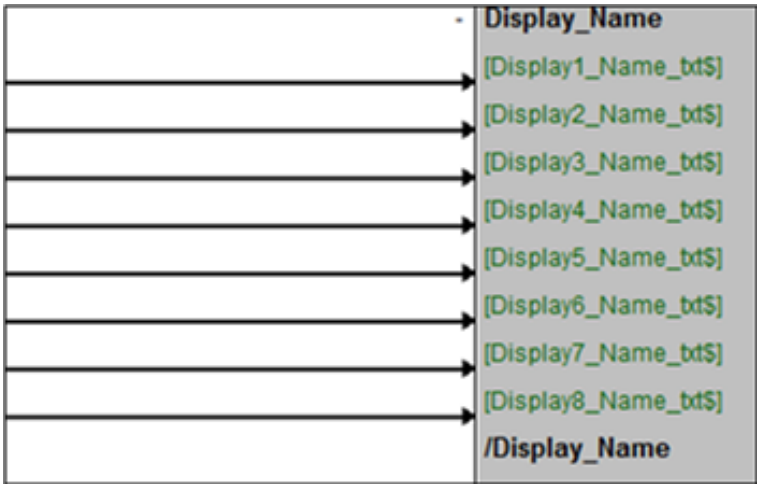
1. Verify that the Fusion SSI Display Usage module has been added to the program as described in Add the Fusion SSI Module to a Program (on page 16).
2. Program the three input lines according to the following guidelines:
   - The digital signal for **Display[x]_Power_On_fb** (where "[x]" is the display number) should latch the signal high when the display is on.

     Program View



- Determine whether the display is a Crestron Connected® display.
  - If yes, connect the Power On feedback signal to the Fusion SSI Display Usage module for the corresponding display.
  - If no, connect the Power On command signal directly to the Fusion SSI Display Usage module.
- Determine whether the display is controlled by TCP/IP, RS-232, or RS-422.
  - If yes and the display has Power On feedback, connect the Power On feedback signal to the Fusion SSI Display Usage module for the corresponding display.
  - If no, connect the Power On command signal to the Fusion SSI Display Usage module for the corresponding display.
- Determine whether the display is controlled by discrete IR.
  - If yes, use the discrete power on and power off IR signals to drive the **Display[x]_Power_On_fb** signal using a latch or similar logic.
  - If no, it is not possible to provide accurate usage information to Crestron Fusion. A different display is required to provide this information.
- To send the display name via programming (overrides parameter field at bottom of module), send a serial string containing the name to **Display[x]_Name_txt$** (for example, LED Left Display or LED Right Display).

  Display(x)_Name_txt$

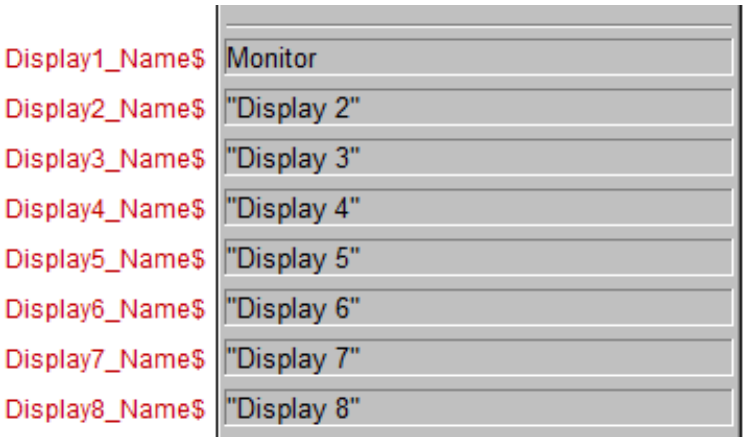


3. If desired, program the optional **Restart_Counters** signal to zero on the rising edge. For example, pulse this signal at midnight so that the device usage data is specific for the day and does not overlap to the next.
4. If the Fusion SSI Display Usage module is full, add another module to the program and repeat these steps, as necessary.

5. If the individual lines described above for display name are not used, fill in the parameter fields listed at the bottom of the module.

   The **Display[x]_Name$** should be filled in according to the name of the display. This field corresponds to the display input fields at the top of the module.

   Display(x)_Name$



NOTES:

- Display usage is calculated by the rise and fall of the digital signal for **Display_[x]_Power_On_fb**. Based on the incoming digital signal, the rising edge starts the timer and the falling edge of the digital signal stops the timer. The module calculates the time (in minutes) and converts the data to be sent to the Crestron Fusion database for storage.
- The **DisplayUsage** analog signal shows only the current total display usage reported by the display.

Signal Definitions

Refer to the following tables for Fusion SSI Display Usage module signal definitions.

Fusion SSI Display Usage Module Inputs

Module Signal Name	Definition
Display[x]_Power_On_fb	This signal links to the logic on the display's control module for Power On. [x] = the display number.
[Display[x]_Name$]	This signal defines the name of the display, which overrides any value set in the Display[x]_Name$ parameter field. [x] = the display number.
Restart_Counters	On the rising edge of this signal, all active display usage counters are stopped, active sessions are stopped and recorded, and counters are set to zero.

Fusion SSI Display Usage Module Outputs

Module Signal Name	Definition
Display_Usage_String$	This signal contains the data created by the module outputs on this serial string and is connected to the Display_Usage_String$ input on the Crestron Fusion SSI module.
Display_[x]_Power_ Status$	This signal outputs the current status of the display power as a string, either "Powered On" or "Powered Off".
Display_[x]_Name_ Current$	This signal contains the name of the display based on the corresponding Display Name parameter or the name that overwrites the parameter when supplied on the corresponding Display Name input string.

Fusion SSI Display Usage Module Parameters

Module Signal Name	Definition
Display[x]_Name$	This signal contains the data created by the module outputs on this serial string and is connected to the Display_Usage_String$ input on the Crestron Fusion SSI module. [x] = the display number.

Program the Fusion SSI Device Usage Module

Crestron Fusion allows users to track usage information for AV devices in the room such as DVD players, PCs, and document cameras. The Fusion SSI Device Usage module generates and formats usage data (in minutes) that is sent to the database. The information is available in the Crestron Fusion reports section.

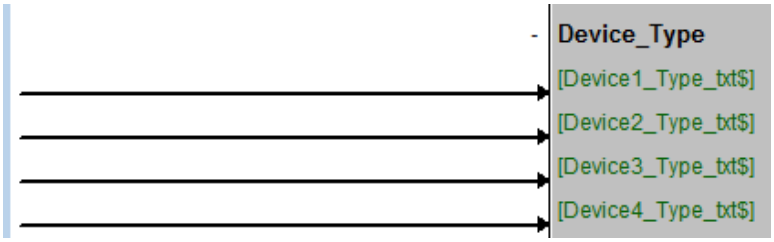
NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to implementing the Fusion SSI Device Usage module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

Programming Procedure

To program the Fusion SSI Device Usage module:

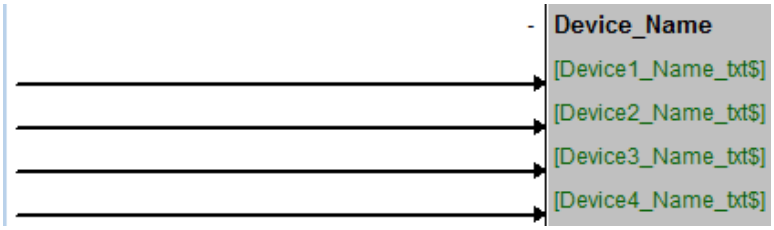
1. Verify that the Fusion SSI Device Usage module has been added to the program as described in Add the Fusion SSI Module to a Program (on page 16).
2. If desired, program the optional **Restart_Counters** signal to zero on the rising edge. For example, pulse this signal at midnight so that the device usage data is specific for the day and does not overlap to the next or pulse at a meeting start to produce a specific usage session for that meeting.
3. Program the three input lines according to the following guidelines:
   - To send the device type via programming, send a serial string containing the type to **Device_[x]_Type_txt$** where "[x]" = the device number (for example, PC). The device type string should follow SSI guidelines outlined previously in this document.

     Device Type Examples



- To send the device name via programming, send a serial string containing the name to **Device_[x]_Name_txt$** where "[x]" = the device number (for example, Instructor DVD Player). The device name string should follow SSI guidelines.

  Device Name Examples



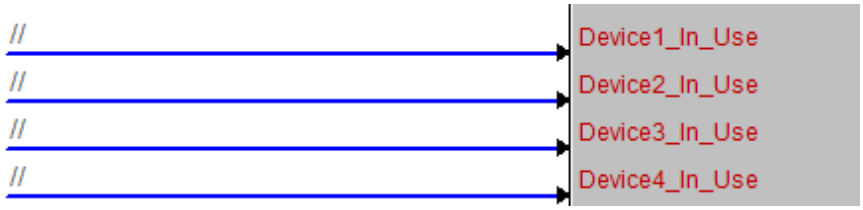
- The digital signal for **Device_[x]_In_Use** is used to calculate the usage data. Programming logic should be written to drive this signal high when the device is in use and low when it is not in use. Follow the steps below:

  NOTE: The device in use may be determined from a combination of conditions, which includes the conditions listed below and the device must be the active source in the room. Device usage indicates that the device is on, sync is present, and the device is the active source on one of the displays.

  - If the device has a button (such as a Play button) that starts the device, use the feedback from this signal to drive the digital input high. If feedback does not get held high on the press, use a logic symbol such as a Set/Reset Latch or Toggle to hold the digital input high for as long as the device is in use.
  - If the device has power feedback for on and off, use the feedback from Power On to drive the input high and then low when the power is turned off.
  - If feedback from the device is not available, use the feedback from an interlock symbol (interlocked with other devices) to drive the digital input for Device_[x]_In_Use high and low.

- The **Device_[x]_Type$** signal should be filled in according to the type of the device. This field corresponds to the device input fields at the top of the module. Refer to Approved Asset Types (on page 9) for recommended device types.

  Device_[x]_Type $ Example



- The **Device[x]_Name$** signal should be filled in according to the name of the device. This field corresponds to the device input fields at the top of the module.

  Device_[x]_Name$ and Device_[x]_Type$ Examples

Device1_Type$	Generic Source
Device1_Name$	AM-100
Device2_Type$	Generic Source
Device2_Name$	"Device 2"
Device3_Type$	Generic Source
Device3_Name$	"Device 3"
Device4_Type$	Generic Source
Device4_Name$	"Device 4"

4. If the Fusion SSI Device Usage module is full, add another Fusion SSI Device Usage module to the program and repeat these steps as necessary.

NOTES:

- Device Usage is calculated by the rise and fall of the digital signal for **Device_[x]_In_Use**. Based on the incoming digital signal, the rising edge starts the timer and the falling edge of the incoming digital signal stops the timer. The module calculates the time (in minutes) and converts the data to be sent to the Crestron Fusion database.
- The custom name and type fields are used for systems where this information is dynamically sent to the program based on preconfigured devices. Two examples of this are the DMPS and DM® switcher configurations. The names of the inputs and outputs can be entered via Crestron Toolbox™ software.

  In programming, these names can be sent via serial strings from the switcher or DMPS to this module. It is suggested to standardize the programming across all room standards, although the names in each room for sources may differ.
- All displays should be programmed to use the Display Usage module separately and exclusively of the Device Usage module.

Signal Definitions

Refer to the following tables for Fusion SSI Device Usage module signal definitions.

Fusion SSI Device Usage Module Inputs

Module Signal Name	Definition
Restart_Counters	On the rising edge of this signal, all active display usage counters are stopped, active sessions are stopped and recorded, and counters are set to zero.
Device_[x]_In_Use	This signal triggers the module counter to start for the source equipment on the rising edge and turns it off on the falling edge. [x] = The device number.
Device_[x]_Type_txt$	This signal defines the type of source equipment. [x] = The device number.
Device_[x]_Name_txt$	This signal defines the actual device name for the source equipment. [x] = The device number.

Fusion SSI Device Usage Module Outputs

Module Signal Name	Definition
Device_Usage_String$	This signal contains the data created by the module outputs on this serial string and is connected to the Device_Usage_String_in$ input on the Fusion SSI module.

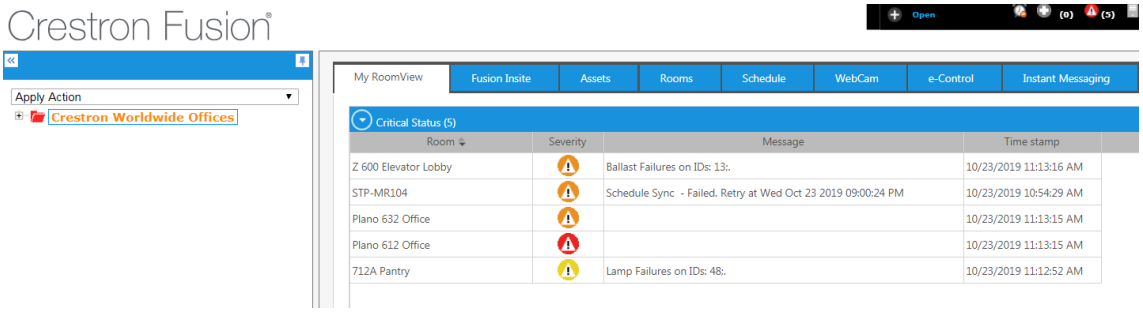
Fusion SSI Device Usage Module Parameters

Module Signal Name	Definition
Device_[x]_Type$	This signal defines the type of source equipment. [x] = the device number.
Device_[x]_Name$	This signal defines the specific name of the source equipment. [x] = the device number.

Program the Fusion SSI Equipment Status Module

System notifications are triggered by programming and sent to Crestron Fusion to report an incident, including a level of severity and a custom message. These messages and their corresponding severity levels are displayed in the **My RoomView** tab in the Crestron Fusion web client. A new incoming message overwrites the previous message, so only the most current message is displayed for the room in Crestron Fusion. All messages are logged (if logging is enabled for the ErrorMessage attribute) and can be viewed by running the built-in Crestron Fusion report "Error Alerts".

Crestron Fusion Web Client - My RoomView Tab



The Fusion SSI Equipment Status module is designed to send trigger notifications from programming to Crestron Fusion.

NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to implementing the Fusion SSI Equipment Status module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

Programming Procedure

To program the Fusion SSI Equipment Status module:

1. Verify that the Fusion SSI Device Usage module has been added to the program as described in Add the Fusion SSI Module to a Program (on page 16).
2. Program the input line according to the guidelines below. Up to 16 status messages can be programmed on this module to be sent to Crestron Fusion.
    - The first section focuses on setting up the actual triggers.
        - Status_Trigger_[x] (where "[x]" is the trigger number) should be tied to programming logic that generates an error state via a digital high/low state.

Status_Trigger_[x]

  - The corresponding message (defined below) is sent on the rising edge of the digital signal.
  - The message is cleared on the falling edge of the signal.
- The **Trigger_[x]_Severity** indicates the severity level of the message defined by the drop-down menu. The selections are:
  - Ok (0d)
  - Notice (1d)
  - Warning (2d)
  - Error (3d)
  - Fatal (4d)

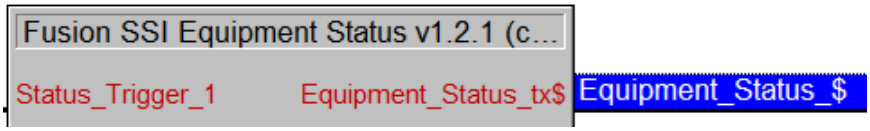
    NOTE: The Ok severity is automatically triggered on the falling edge to clear the message in Crestron Fusion. Selecting this severity does not show in Crestron Fusion but does record to the database.
- The **Trigger_[x]_Status_Error_Text** indicates the actual message. Fill in this field with the message to be sent to Crestron Fusion, for example, Device is Offline, Temperature To High, Motion Detected.
- The **Trigger_[x]_Ok_Text** indicates the message is sent when the status changes to Ok (assigned analog value of 0d).

Triggers

Trigger_1_Severity	Fatal
Trigger_1_Status_Error_Text	Display Offline
Trigger_1_Ok_Text	Display Online
Trigger_2_Severity	Error
Trigger_2_Status_Error_Text	AM-100 Offline
Trigger_2_Ok_Text	AM-100 Online
Trigger_3_Severity	Fatal
Trigger_3_Status_Error_Text	DGE-100 Offline
Trigger_3_Ok_Text	DGE-100 Online
Trigger_4_Severity	Error
Trigger_4_Status_Error_Text	TS Offline
Trigger_4_Ok_Text	TS Online

- The **Equipment_Status_tx$** should be tied to the **Equipment_Status$** on the Crestron Fusion SSI module.

Equipment Status



Signal Definitions

Refer to the following tables for Fusion SSI Equipment Status module signal definitions.

Fusion SSI Equipment Status Module Inputs

Module Signal Name	Definition
Status_Trigger_[x]	This signal triggers the coordinating parameter fields in a serial string format to be sent to the Crestron Fusion SSI module. [x] = The trigger number.

Fusion SSI Equipment Status Module Outputs

Module Signal Name	Definition
Equipment_Status_tx$	This serial string output is sent to the Device_Usage_String_in$ input on the Crestron Fusion SSI module.

Fusion SSI Equipment Status Module Parameters

Module Signal Name	Definition
Prepend_Severity_Text	This parameter, when set to Yes, will prepend the severity text string to the message (Ok, Notice, Warning, Critical, Fatal). This is useful when using Regex for email alerts and helps with some reports.
Trigger_[x]_Severity	This parameter defines the severity level of the status being sent. The values can be Ok, Notification, Warning, Critical, or Fatal. [x] = The trigger number.
Trigger_[x]_Status_ Error_Text	This parameter defines the custom message being sent. [x] = The trigger number.
Trigger_[x]_Ok_Text	This parameter defines the custom message to be sent when the trigger resets to Ok. This occurs when the Status_Trigger_[x] digital signal goes low. [x] = The trigger number.
Time_Before_Status_ Trigger	This signal is a gate to help prevent false positives. The default value is 120s.

Program the Fusion SSI Help Request and Response Module

The Fusion SSI Help Request and Response module is used to trigger help requests from the room, which are then sent to Crestron Fusion. Responses are sent to the room and typically are displayed on a touch screen or display for the end user. This module also accommodates the ability to send and receive custom messages.

NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to implementing the Fusion SSI Help Request and Response module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

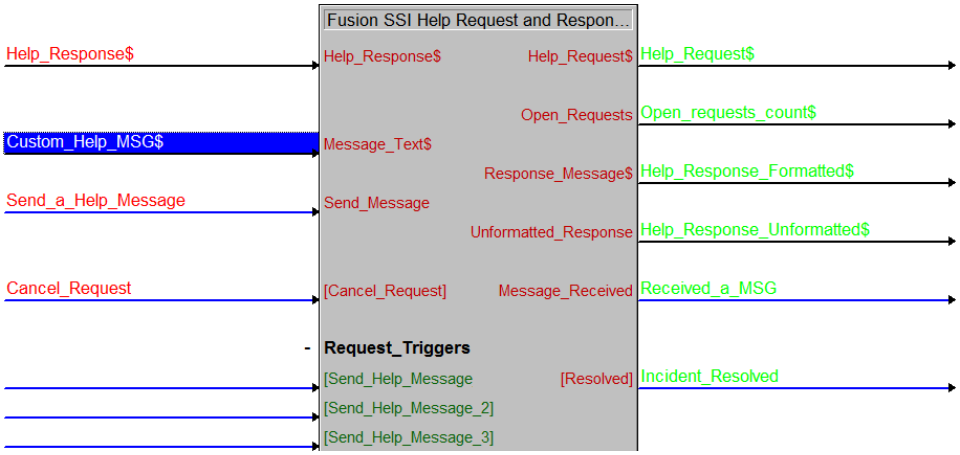
Programming Procedure

To program the Fusion SSI Help Request and Response module:

1. Verify that the Fusion SSI Device Usage module has been added to the program as described in Add the Fusion SSI Module to a Program (on page 16).
2. Program the inputs and outputs of the module:
   - The **Help_Response$** input signal is tied to the Fusion SSI Module on the **[Help_Response$]** output.
   - The **Message_Text$** should be tied to a serial signal providing formatted text for a custom help request. This can be invoked by using a built-in keyboard on the touch screen or a custom keyboard page designed in the user interface project.
   - The **Send_Message** digital input works with the serial **Message_Text$**. In order to send the message to Crestron Fusion, send the serial string to **Message_Text$** and then immediately follow it with sending the **Send_Message** digital input high. The **Message_Text$** enters a serial queue internally to the module and is then sent with the rising edge of **Send_Message**.
   - The **[Cancel_Request]** digital input should be tied to a button on the user interface so that, when sent high, it clears the help request coming from the room.
   - The **[Send_Help_Message_[x]]** digital inputs correspond to the **Message [x]** parameters. The digital signal is pulsed high for the corresponding message to be sent to the **Help_Request$** output. These digital inputs should tie to button presses on the user interface.
   - The **Help_Request$** serial output signal ties to the Fusion SSI Module on the **[Help_Request$]** serial input.
   - The **Open_Requests** is an output that outputs the ID numbers of any open requests.

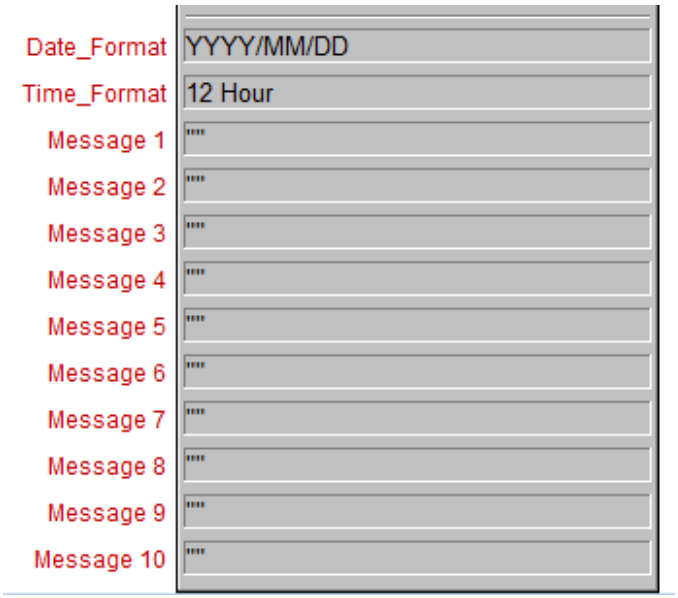
- The **Response_Message$** serial output is tied to a serial input on a user interface. This displays the response message coming from Crestron Fusion.
- The **Unformatted_Response** output displays the response message coming from Crestron Fusion similar to **Response_Message$**, but without the formatting that includes the date/time stamp and the from and to fields.
- The **Message_Received** digital output can be used to trigger subpage references or other methods to invoke a screen to appear that shows the Help conversation.
- The **[Resolved]** digital output should be tied to the second SIO to end and clear the conversation. The resolved trigger is a button press from Crestron Fusion, which is then processed through the **Help_Response$ XML**.

Help_Response$ and Help_Request$ Example



- The parameter fields dictate the formatting of the time, date, and preset messages, which are triggered by **[Send_Help_Message_[x]]** correspondingly.

Help_Response$ and Help_Request$ Example



Signal Definitions

Refer to the following tables for Fusion SSI Help Request and Response module signal definitions.

Fusion SSI Help Request and Response Module Inputs

Module Signal Name	Definition
Help_Response$	This serial string is returned from Crestron Fusion. This module parses this string and formats it for the user interface. This string is in XML format.
Message_Text$	This serial string is sent to the Crestron Fusion SSI module and passed through to Crestron Fusion. The maximum character length is 250 characters and is not gathered or buffered.
Send_Message	This digital signal (when sent high) checks for the serial string in the Message_Text$ or in a continued text conversation format from the current text serial string and sends the string to the Crestron Fusion SSI module and then passes it through to Crestron Fusion.
[Cancel_Request]	This signal checks for existing conversation and sends a cancel to close the conversation.
[Send_Help_ Message_[x]]	This digital signal (when triggered) sends out the corresponding Message [x] parameter field to the Crestron Fusion SSI module and passes it through to Crestron Fusion.

Fusion SSI Help Request and Response Module Outputs

Module Signal Name	Definition
Help_Request$	This serial string (that has been formatted by the module) is sent out to the Crestron Fusion SSI module and is then passed through to Crestron Fusion.
Open_Requests	This outputs the ID of any open requests.
Response_Message$	This serial string (that has been formatted by the module) is sent to a user interface. The format has a date and time stamp. A message is formatted as a [From] to [To]: [Message] where the From field contains the user sending the response to the room. The [Message] contains the text string message sent by the user.
Unformatted_ Response	This output is the same as Response_Message$ without any formatting that includes the date/time stamp and the from and to fields.
Message_Received	This digital signal remains high for the duration of the conversation. A resolved command issued from Crestron Fusion or a Cancel Message trigger forces the digital signal to go low.
[Resolved]	This digital signal is triggered 5 seconds after a canceled or resolved command has been issued.

Fusion SSI Help Request and Response Module Parameters

Module Signal Name	Definition
Date_Format	This signal selects the type of date format used for the help request and response conversation. 1 = MM/DD/YYYY 2 = DD/MM/YYYY 3 = YYYY/MM/DD 4 = MM/DD/YY
Time_Format	This signal selects the type of date format used for the help request and response conversation. 1 = 24-Hour Format 2 = 12-Hour Format
Message [x]	This preset message field is sent based on the corresponding [Send_Help_Message_[x]] string.

Program the Fusion SSI Broadcast Message Module

The Fusion SSI Broadcast Message Module is used to handle broadcast messages sent from a Crestron Fusion system. Received messages may be routed to touch screens, displays, and other devices. These messages can be general information, weather, emergency, or any message that needs to be sent quickly and can effectively reach the attention of all employees, students, and so forth.

This module also records received messages and message types to the processor log file.

NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to implementing the Fusion SSI Broadcast Message module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

Programming Procedure

To program the Fusion SSI Broadcast Message module:

1. Verify that the Fusion SSI Device Usage module has been added to the program as described in Add the Fusion SSI Module to a Program (on page 16).
2. Program the inputs and outputs according to the following guidelines:
   - The **Broadcast_Message_Type** and **Broadcast_Message$** inputs are tied to the outputs of **[Broadcast_Message_Type]** and **[Broadcast_Message$]** on the Fusion SSI module respectively.

     Broadcast_Message_Type and Broadcast_Message$ Example

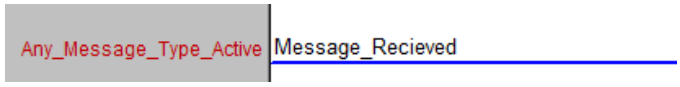
   - The **Message_Type_Value** returns a value of 0–9 and can be used to trigger events in programming.

     Message_Type_Value Example

   - The **Any_Message_Type_Active** can be used in programming to indicate that a message has been received into the program.

     Any_Message_Type_Active Example

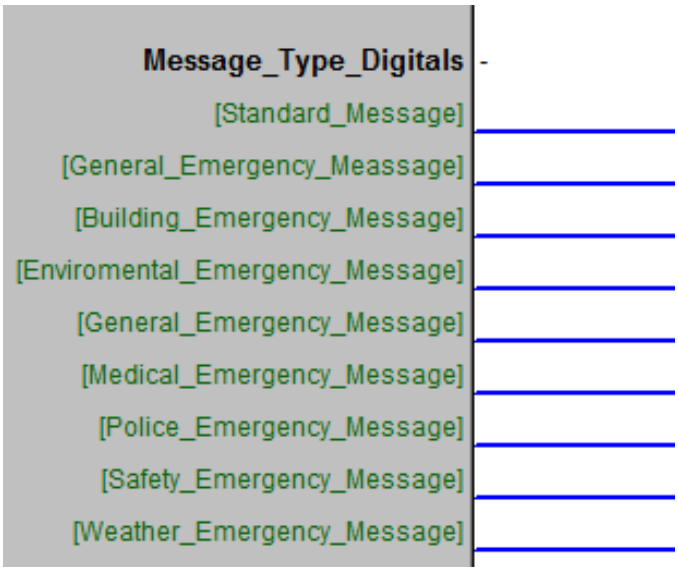
     

- The **Message_Text$** is the actual message sent from Crestron Fusion. Normally this is sent to an indirect text field on an interface.

  Message_Text$ Example

- The **Message_Type_Digitals** can be used to trigger subpages or other logic when that particular message type is selected.

  Message_Type_Digitals Example



Signal Definitions

Refer to the following tables for Fusion SSI Broadcast Message module signal definitions.

Fusion SSI Broadcast Message Module Inputs

Module Signal Name	Definition
Broadcast_Message_Type	This analog value indicates the type of broadcast message that is received from Crestron Fusion. The values are as follows: 0 – Clear Message 1 – Standard Message 2 – Building Emergency Message 3 – Environmental Emergency Message 4 – General Emergency Message 5 – Medical Emergency Message 6 – Police Emergency Message 7 – Safety Emergency Message 8 – Weather Emergency Message 9 – Custom Emergency Message
Broadcast_Message$	This serial string is received and contains the actual message. This string can be sent to an indirect text field to display on a touch screen

Fusion SSI Broadcast Message Module Outputs

Module Signal Name	Definition
Message_Type_Value	This analog value is returned and indicates the message type. The values returned are as follows: 0 – Clear Message 1 – Standard Message 2 – Building Emergency Message 3 – Environmental Emergency Message 4 – General Emergency Message 5 – Medical Emergency Message 6 – Police Emergency Message 7 – Safety Emergency Message 8 – Weather Emergency Message 9 – Custom Emergency Message
Any_Message_Type_ Active	This digital signal indicates that a message has been received. This signal also sends a received message and message type to the processor log file.
Message_Text$	This serial value returns the message received from Crestron Fusion. This value can be sent to an indirect text field to be displayed on a touch screen.
Message_Type_Digitals	These digital values are held high depending on the message type received from Crestron Fusion. The corresponding value holds the output high until the message is cleared. These values can be used to trigger subpages on a touch screen to display message information.

Program the Fusion SSI Text Message Send and Receive Module

The Fusion SSI Text Message Send and Receive module is used to send text messages between the room and Crestron Fusion. This procedure identifies programming text messages that then send the data to the Crestron Fusion SSI module.

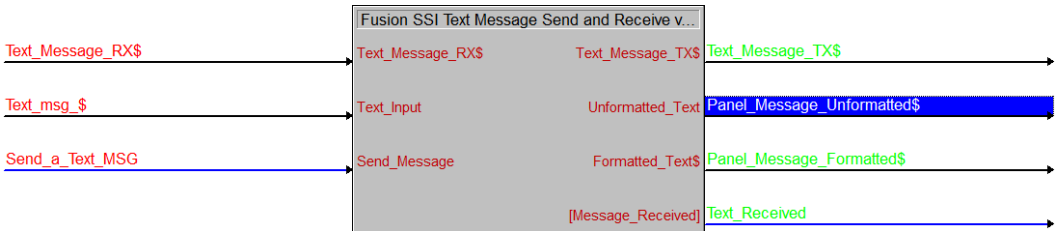
NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to implementing the Fusion SSI Text Message Send and Receive module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

Programming Procedure

To program the Fusion SSI Text Message Send and Receive module:

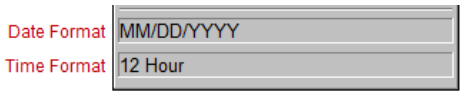
1. Verify that the Fusion SSI Device Usage module has been added to the program as described in Add the Fusion SSI Module to a Program (on page 16).
2. Program the inputs and outputs according to the following guidelines:
   - The **Text_Message_RX$** is tied to the Fusion SSI Module on the **[Text_Message_RX$]** serial output signal.
   - **Text_Input** should be tied to a serial signal providing formatted text to be sent to Crestron Fusion. This can be invoked by using a built-in keyboard on the touch screen or a custom keyboard page designed in the user interface project.
   - The **Send_Message** digital input should be tied to a button on the touch screen to send the text message string.
   - The **Text_Message_TX$** serial output ties to the Fusion SSI module on the **[Text_Message_TX$]** serial input signal.
   - [Message_Received] is a digital output that is pulsed upon a received message. This can be used to trigger a subpage, small graphic, or text field to indicate that a message has been received.

Text Message Send and Receive Example



- The parameter fields dictate the formatting for the time and date stamp associated with the text messages.

  Date Format and Time Format Example



Signal Definitions

Refer to the following tables for Fusion SSI Text Message Send and Receive module signal definitions.

Fusion SSI Text Message Send and Receive Module Inputs

Module Signal Name	Definition
Text_Message_RX$	This serial string is received from Crestron Fusion. This string is formatted by the module and sent out via the Formatted_Text$ serial output.
Text_Input	This serial string contains the text message sent out to Crestron Fusion.
Send Message	This digital input formats the serial string message to be sent to Crestron Fusion and then transmits the string.

Fusion SSI Text Message Send and Receive Module Outputs

Module Signal Name	Definition
Text_Message_TX$	This is the formatted serial string that is sent to Crestron Fusion.
Unformatted Text	This serial output is the Text Message string without any formatting.
Formatted_Text$	This is the serial output of the serial text message string. The data is formatted as: "Date Time [From] to [To]: [Message]", where [From] contains the current Crestron Fusion username who is sending the message, [To] is always the room user, and [Message] contains the actual serial string message sent from Crestron Fusion.
[Message_Received]	This is the digital signal which is pulsed to indicate a message has been received.

Fusion SSI Text Message Send and Receive Module Parameters

Module Signal Name	Definition
Date_Format	This signal selects the type of date format used for the text message conversation. 1 = MM/DD/YYYY 2 = DD/MM/YYYY 3 = YYYY/MM/DD 4 = MM/DD/YY
Time_Format	This signal selects the type of date format used for the text message conversation. 1 = 24-Hour Format 2 = 12-Hour Format

Program the Fusion SSI Scheduling Awareness Module

The Fusion SSI Scheduling Awareness Module is used to retrieve basic scheduling information from Crestron Fusion.

NOTE: The Fusion Room device definition and the Fusion SSI module must be added to the program prior to implementing the Fusion SSI Scheduling Awareness module. For more information, refer to Add the Fusion Room Device Definition (on page 12) and Add the Fusion SSI Module to a Program (on page 16), respectively.

Programming Procedure

To program the Fusion SSI Scheduling Awareness module:

1. Verify that the Fusion SSI Device Usage module has been added to the program as described in Add the Fusion SSI Module to a Program (on page 16).
2. Program the inputs and outputs to retrieve scheduling data from Crestron Fusion, which is then parsed through the module to be sent to a display.

Signal Definitions

Refer to the following tables for Fusion SSI Scheduling Awareness module signal definitions.

Fusion SSI Scheduling Awareness Module Inputs

Signal Group	Module Signal Name	Definition
	Room_Offline_FB	This signal must be tied to offline manager to determine online and offline status. The module behavior is different depending on the response.
Room_Setup	Get_Schedule	This signal can be triggered to send a schedule query for the room.
Room_Setup	Room_Occupied	This signal can be tied to occupancy sensor feedback to state whether the room is occupied or unoccupied.
Room_Setup	Enable_Decline_For_No_Show	This signal (when high) enables a decline for a no show. This cancels meetings that are a preset number of minutes after the meeting start time (if the room has not been occupied or other feedback has not been met) indicating that the room is in use.
Room_Setup	Disable_Push_Registration	This signal disables the request for push notification query from this module.
Meeting_Control	End_Meeting	This signal when sent high ends the current meeting. This does not delete the meeting from the calendar but instead changes the end time of the meeting to make the room available.
Meeting_Control	Extend_Meeting_15_Minutes	When pulsed extends the current meeting 15 minutes if time is available.
Meeting_Control	Extend_Meeting_30_Minutes	This signal (when pulsed) extends the current meeting 30 minutes, if time is available.
Meeting_Control	Extend_Meeting_60_Minutes	This signal (when pulsed) extends the current meeting 60 minutes, if time is available.

Fusion SSI Scheduling Awareness Module Outputs

Signal Group	Module Signal Name	Definition
Room_Setup	Room_Name_txt$	This signal returns the name of the room.
Room_Setup	Get_Schedule_Busy_fb	When high this signal indicates that the room is currently waiting on scheduling data.
Room_Setup	Decline_For_No_Show_Enabled_fb	This signal indicates if Decline for No Show has been enabled.

Signal Group	Module Signal Name	Definition
Room_Setup	Room_Push_ Registered_fb	This signal indicates if the room has been push registered. Relies on feedback from Crestron Fusion to indicate this.
UI_Control	[Show_Upcoming_ Hours_and_Minutes_ Countdown]	This signal goes high if an hour value and a minute value of remaining time until the next scheduled meeting are present.
UI_Control	[Show_Upcoming_ Minutes_Countdown]	This signal goes high if only a minute value of remaining time until the next scheduled meeting is present.
UI_Control	[Upcoming_ Countdown_Hours_ Value]	This signal indicates the hours remaining until the next scheduled meeting.
UI_Control	[Upcoming_ Countdown_Minutes_ Value]	This signal indicates the minutes remaining until the next scheduled meeting.
UI_Control	[Show_Current_ Minutes_Countdown]	This signal goes high if only a minute value of remaining time for a current meeting is present.
UI_Control	[Show_Current_ Hours_and_Minutes_ Countdown]	This signal goes high if an hour value and a minute value of remaining time for a current meeting are present.
UI_Control	[Current_ Countdown_Hours_ Value]	This analog value is returned with the hour countdown value for the current meeting.
UI_Control	[Current_ Countdown_Minutes_ Value]	This analog value is returned with the minute countdown value for the current meeting.
UI_Control	[Show_Available_for_ the_Next_Banner]	This digital signal triggers high when the room is available. This trigger is used to indicate time until the next meeting.
UI_Control	[Show_Available_ For_the_Day_Banner	This digital signal triggers high when the room is available. This trigger is used when there are no meetings for the rest of the day.
UI_Control	[Show_Reserved_ Banner]	This digital signal triggers high when the room is currently reserved. This signal can be used to display a message on the user UI when the room is reserved.
UI_Control	[Show_Info_Field]	This digital signal triggers high when the room is currently reserved to show meeting information. This signal can be used to display a subpage on the user interface to show

Signal Group	Module Signal Name	Definition
		current meeting information.
UI_Control	[Show_Next_Up]	This digital signal triggers high, when the room is currently reserved, to show meeting information for the next meeting. This signal can be used to display a subpage on the user interface to show the current meeting information.
Meeting_Control	Extend_Meeting_ Busy_fb	This digital signal (when high) indicates the system is currently busy updating the meeting to extend it the predetermined amount of time.
Meeting_Control	Allow_Extend_ Meeting_15_Minutes_ fb	This is the digital pulsed feedback that indicates the extend meeting 15 minutes digital input has been selected.
Meeting_Control	Allow_Extend_ Meeting_30_ Minutes_fb	This is the digital pulsed feedback that indicates the extend meeting 30 minutes digital input has been selected.
Meeting_Control	Allow_Extend_ Meeting_60_ Minutes_fb	This is the digital pulsed feedback that indicates the extend meeting 60 minutes digital input has been selected.
Meeting_Control	Extend_Meeting_ Success_Pulse	This is the digital pulsed feedback that indicates the meeting has successfully been extended.
Meeting_Control	Extend_Meeting_ Error_Pulse	This is the digital pulsed feedback that indicates the meeting has not been extended due to an error.
Current_Meeting_ Information	Meeting_In_ Progress_fb	This is the digital feedback that indicates a meeting is currently in progress.
Current_Meeting_ Information	Current_Meeting_ Subject_txt$	This signal contains data for the current meeting subject name.
Current_Meeting_ Information	Current_Meeting_ Organizer_txt$	This signal contains data for the current meeting organizer name.
Current_Meeting_ Information	Current_Meeting_ Start_Time_txt$	This signal contains data for the current meeting start time.
Current_Meeting_ Information	Current_Meeting_ End_Time_txt$	This signal contains data for the current meeting end time.
Current_Meeting_ Information	Current_Meeting_ Formatted_Time_ txt$	The data contained in this signal is a formatted string indicating the date and start time through the end time of the current meeting, for example, 01-01-01 9:30-10:30.

Signal Group	Module Signal Name	Definition
Current_Meeting_ Information	Current_Meeting_ Time_Remaining_txt$	This signal contains data for the time remaining for the current meeting.
Current_Meeting_ Information	Current_Meeting_ RV_Meeting_ID_txt$	This signal contains data for the current meeting ID in the Crestron Fusion database.
Current_Meeting_ Information	Current_Meeting_ Start_Time_fb	This analog value can be used for a gauge to graphically indicate the countdown time before the current meeting start time.
Current_Meeting_ Information	Current_Meeting_ End_Time_fb	This analog value can be used for a gauge to graphically indicate the countdown time before the current meeting end time.
Current_Meeting_ Information	Current_Meeting_ Time_Remaining_fb	This analog value can be used for a gauge to graphically indicate the countdown time remaining for the current meeting.
Next_Meeting_ Information	Next_Meeting_ Subject_txt$	This signal contains data for the next meeting's subject name.
Next_Meeting_ Information	Next_Meeting_ Organizer_txt$	This signal contains data for the next meeting's organizer name.
Next_Meeting_ Information	Next_Meeting_Start_ Time_txt$	This signal contains data for the next meeting's start time.
Next_Meeting_ Information	Next_Meeting_End_ Time_txt$	This signal contains data for the next meeting's end time.
Next_Meeting_ Information	Next_Meeting_ Formatted_Time_ txt$	The data contained in this signal is a formatted string indicating the date and start time through the end time of the next meeting, for example, 01-01-01 9:30-10:30.
Next_Meeting_ Information	Next_Meeting_RV_ Meeting_ID_txt$	This signal contains data for the next meeting ID in the Crestron Fusion database.
Next_Meeting_ Information	Next_Meeting_Start_ Time_fb	This analog value can be used for a gauge to graphically indicate the countdown time before the next meeting start time.
Next_Meeting_ Information	Next_Meeting_End_ Time_fb	This analog value can be used for a gauge to graphically indicate the countdown time before the next meeting end time.
Third_Meeting_ Information	Third_Meeting_ Subject_txt$	This signal contains data for a third meeting's subject name. This meeting is third in order starting with the current meeting as number one.
Third_Meeting_ Information	Third_Meeting_ Organizer_txt$	This signal contains data for the third meeting's organizer name. This meeting is third in order starting with the current meeting

Signal Group	Module Signal Name	Definition
		as number one.
Third_Meeting_ Information	Third_Meeting_Start_ Time_txt$	This signal contains data for the third meeting's start time. This meeting is third in order starting with the current meeting as number one.
Third_Meeting_ Information	Third_Meeting_End_ Time_txt$	This signal contains data for the third meeting's end time. This meeting is third in order starting with the current meeting as number one.
Third_Meeting_ Information	Third_Meeting_ Formatted_Time_ txt$	This signal contains data in a formatted string and indicates the date and start time through the end time of the third meeting, for example, 01-01-01 9:30-10:30.
Third_Meeting_ Information	Third_Meeting_RV_ Meeting_ID_txt$	This signal contains data for the third meeting ID in the Crestron Fusion database. This meeting is third in order starting with the current meeting as number one.
Third_Meeting_ Information	Third_Meeting_Start_ Time_fb	This analog value can be used for a gauge to graphically indicate the countdown time before the third meeting start time.
Third_Meeting_ Information	Third_Meeting_End_ Time_fb	This analog value can be used for a gauge to graphically indicate the countdown time before the third meeting end time.

Fusion SSI Scheduling Awareness Module Parameters

Module Signal Name	Definition
Decline_For_No_Show_ Minutes	This signal dictates the number of minutes to wait past the meeting start time before sending a meeting cancellation that is due to no attendee presence in the room.
Allow_Push_ Registration	This signal enables and disables the Push Registration.
Upcoming_Show_ Minutes	This signal indicates the character string length for the upcoming show minutes serial string.
Scheduler ID	This signal indicates the crosspoint ID for a Fusion SSI Scheduling Awareness module and is required to connect the Fusion SSI Scheduling Awareness module to the main Fusion SSI module. The value must match the Scheduling ID parameter on the main Fusion SSI module. If the values do not match, the Fusion SSI Scheduling Awareness module will not request the message body contained in the scheduled events from Crestron Fusion.

Crestron Fusion Request/Response XML Streams

This section of the document describes the XML formats for certain Crestron Fusion communication, including schedule manipulation, time sync, attribute read/write, and others. This is a subset that is related to the Fusion SSI module.

The Fusion SSI module for SIMPL implements these methods, but SIMPL Sharp programmers may need to implement these methods directly.

The procedures in this section follow these assumptions:

- The term "Crestron Fusion" will be used below to refer to Crestron Fusion RV or Crestron Fusion EM. Where a distinction is required, the full name will be used.
- All dates will adhere to international formatting: yyyy-mm-dd (for example, "2019-05-20").
- All times will utilize military nomenclature (for example, "15:30:00").
- If a RoomID is not sent with the request, Crestron Fusion will assume the request is associated with the local room assigned to the symbol in RoomView® software.

Basics

Crestron Fusion exchanges requests and responses between itself and a connected device (processor or touch screen) via a device extender, which permits serial data to flow in either direction. The typical exchange is that a request is sent to Crestron Fusion, and a response is delivered from Crestron Fusion.

In some cases, an unsolicited response may be sent by Crestron Fusion to the device (such as push notifications for Schedule Response or when a room is changed in the database for Room Configuration Response). In this case, there is no follow-up response from the device.

All data flows are in XML format and are described in this document.

NOTE: Some earlier versions of Crestron Fusion (9.3 and below) will deliver ill-formed XML in certain cases. To support these older versions, the device code must handle exceptions from its XML parser.

Data Fields

The following fields appear throughout the XML definitions.

XML Data Fields

Field Name	Data Format	Purpose	Example
RoomID	GUID	Identifies a room	22403c59-8bc7-488d-a4f0-a5474faa6d75
RequestID	Opaque string[1]	Permits a caller to identify a unique request sent to Crestron Fusion	User12345
MeetingID	GUID	Identifies a meeting (appointment)	78498c77-28d8-4942-a375-86d3425362ca
HourSpan	Number	A number of hours greater than zero (as of Crestron Fusion 10, this may be a floating point number or an integer.	24
DateTime	RFC 3339 date and time	A date and time in RFC 3339 format, which is always assumed to be a local time to the device	2019-04-30T13:13:13
Field	Field name	One of the Schedule Response field names as listed in Appendix A: Filtered Schedule Request Fields (on page 79).	MeetingID

[1] This term implies that the value sent by the device is not examined by Crestron Fusion, and only has meaning to the device.

Occasionally, there are side comments within the XML box that provide information and are formatted as follows: [example]. These do not appear in the XML streams.

Schedule Requests

Schedule requests are used to request appointments for a room. Schedule requests can be configured to return all data or specific data, which are referred to as "full" and "filtered" requests, respectively. Additionally, requests can be made for a single calendar instead of all events by providing the **MeetingID** field.

Full Schedule Request

The following request returns all scheduling data for all events over the defined date and time period.

```
<RequestSchedule>
   <RequestID>RequestID</RequestID>
   <RoomID>RoomID</RoomID>
   <Start>DateTime</Start>
   <HourSpan>HourSpan</HourSpan>
   <SMTP>AlternateSmptAddress</SMTP>
</RequestSchedule>
```

Filtered Schedule Request (All Meetings)

The following request returns a filtered schedule of all events over the defined data and time, which is based on the supplied parameters. Refer to Appendix A: Filtered Schedule Request Fields (on page 79) for a list of allowed fields.

```
<RequestSchedule>
   <RequestID>RequestID</RequestID>
   <RoomID>RoomID</RoomID>
   <Start>DateTime</Start>
   <HourSpan>HourSpan</HourSpan>
   <SMTP>AlternateSmptAddress</SMTP>
   <FieldList>
      <Field>Field</Field>
      <Field>Field</Field>
      <Field>Field</Field>
      [etc.]
      </FieldList>
 </RequestSchedule>
```

If the <SMTP> tag is not provided, then the email address of the room itself is used. This is the most common use of schedule request. When <SMTP> is used, the contents of the tag should contain an alternate SMTP address to which the Crestron Fusion services are permitted read access.

Filtered Schedule Request (Specific Meeting)

The following request returns a filtered schedule of a specific event. Since the **MeetingID** field is unique to a meeting event, no date or time data is needed for the request. Refer to Appendix A: Filtered Schedule Request Fields (on page 79) for a list of allowed fields.

```
<RequestSchedule>
   <RequestID>RequestID</RequestID>
   <RoomID>RoomID</RoomID>
   <MeetingID>MeetingID</MeetingID>
   <FieldList>
      <Field>Field</Field>
      <Field>Field</Field>
      <Field>Field</Field>
      [etc.]
      </FieldList>
 </RequestSchedule>
```

Schedule Response

The following XML schema is provided as a sample response after requesting a schedule.

- **Recurring** is a tag that identifies whether the event is part of a series.
- **MeetingEnd** is the date and time of the last meeting in a series. The format should be in UTC.
  - If the series has no end, the text "None" should be returned.
  - If the event is not part of a series, the tag should contain no text.

```
<ScheduleResponse>
   <RequestID>152</RequestID>
   <RoomID>nnnnnnnnnn</RoomID>
   <RoomName>Room 10123</RoomName>
   <Event>
      <MeetingID>{150 Char Exchange ID}</MeetingID>
      <Recurring>True</Recurring>
      <InstanceID>789012</InstanceID>
      <dtStart>2006-06-10T09:00:00Z</dtStart>
      <dtEnd>2006-06-10T09:45:00Z</dtEnd>
      <Organizer>John Doe</Organizer>
      <Attendees>
         <Required>
            <Attendee>Tom Brown</Attendee>
            <Attendee>Jane Doe</Attendee>
         </Required>
         <Optional>
            <Attendee>Tom Thumb</Attendee>
            <Attendee>Jane Kits</Attendee>
         </Optional>
      </Attendees>
      <Resources>
         <Rooms>
            <Room>
               <Name>Room 100</Name>
               <ID>123456789</ID>
               <MPType>A</MPType>
            </Room>
            <Room>
               <Name>Room 101</Name>
               <ID>1234</ID>
               <MPType>V</MPType>
            </Room>
         </Rooms>
      </Resources>
      <IsEvent>0</IsEvent>
      <IsRoomViewMeeting>1</IsRoomViewMeeting>
      <IsPrivate>1</IsPrivate>
      <IsExchangePrivate>1</IsExchangePrivate>
      <MeetingTypes>
```

```
            <MeetingType ID=PRINCIPAL Value=True/>
            <MeetingType ID=INROOMDISC Value=True/>
            <MeetingType ID=INROOMPRES Value=True/>
            <MeetingType ID=VCONF Value=True/>
            <MeetingType ID=ACONF Value=True/>
            <MeetingType ID=LIVEMTG Value=True/>
         </MeetingTypes>
         <ParticipantCode>123435</ParticipantCode>
         <PhoneNo>1234567890</PhoneNo>
         <WelcomeMsg>Hello</WelcomeMsg>
         <Subject>Subject</Subject>
         <LiveMeeting>
            <URL></URL>
            <ID></ID>
            <Key></Key>
            <Subject></Subject>
         </LiveMeeting>
         <ShareDocPath>c:\my shared docs\test.txt</ShareDocPath>
         <HaveAttendees>True</HaveAttendees> (only returned on Filtered
         Request responses)
         <HaveResources>False</HaveResources> (only returned on Filtered
         Request responses)
      </Event>
   </ScheduleResponse>
```

Meeting End Time Change Request

The following XML schema is used to request a meeting end time change.

```
<RequestAction>
   <RequestID>RequestID</RequestID>
   <RoomID>RoomID</RoomID>
   <ActionID>MeetingChange</ActionID>
   <Parameters>
      <Parameter ID=”MeetingID” Value=”nnnnn” />”
      <Parameter ID=”EndTime” Value=”0” />”
   </Parameters>
</RequestAction>
```

NOTE: If **EndTime** is set to zero, then the end time will be truncated to the current date and time. If **EndTime** is "*nn*", then the meeting will be extended from the current time by *nn* minutes.

Meeting End Time Change Response

The following XML schema are provided as sample responses after requesting a meeting end time change.

Successful change:

```
<ActionResponse>
   <RequestID>string</RequestID>
   <ActionID>MeetingChange</ActionID>
   <Parameters>
      <Parameter ID=”MeetingID” Value=”nnnnn” />”
      <Parameter ID=”Status” Value=”1” />”
      <Parameter ID=”EndTime” Value=”” />”
   </Parameters>
</ActionResponse>
```

Unsuccessful truncate:

```
<ActionResponse>
   <RequestID>string</RequestID>
   <ActionID>MeetingChange</ActionID>
   <Parameters>
      <Parameter ID=”MeetingID” Value=”nnnnn” />”
      <Parameter ID=”Status” Value=”0” />”
      <Parameter ID=”EndTime” Value=”” />”
   </Parameters>
</ActionResponse>
```

Room List Request

Room list requests are used to obtain a list of rooms in the system. The Location tag is optional and is used to filter rooms:

- If the **Location** tag is left empty, then all rooms are returned.
- If the **Location** tag is used, the value must be an exact match to the value stored in Crestron Fusion for that room.

```
<RequestRoomList>
   <RequestID>152</RequestID>
   <Property>
      <RoomName>Building 100</RoomName
   </Property>
</RequestRoomList>
```

```
<RequestRoomList>
   <RequestID>152</RequestID>
   <Property>
      <RoomName>Conf 1045/300</RoomName
   </Property>
</RequestRoomList>
```

Room List Response

The following XML schema is provided as a sample response after requesting a room list.

```
<RequestRoomResponse>
   <RequestID>152</RequestID>
   <Room>
      <RoomID>nnnnn</RoomID>
      <RoomName>Room 11300</RoomName>
      <Location>Building 11</Location>
      <OnlineStatus></OnlineStatus>
   </Room>
   <Room>
      <RoomID>nnnnn</RoomID>
      <RoomName>Room 11301</RoomName>
      <Location>Building 11</Location>
      <OnlineStatus></OnlineStatus>
   </Room>
   <Room>
      <RoomID>nnnnn</RoomID>
      <RoomName>Room 12110</RoomName>
      <Location>Building 12</Location>
      <OnlineStatus></OnlineStatus>
   </Room>
   <Room>
      <RoomID>nnnnn</RoomID>
      <RoomName>Room 12120</RoomName>
      <Location>Building 12</Location>
      <OnlineStatus></OnlineStatus>
   </Room>
</RequestRoomResponse>
```

Room Configuration Request

Room configuration requests are used to obtain configuration information for a room within the system.

```
<RequestRoomConfiguration>
   <RequestID>152</RequestID>
</RequestRoomConfiguration>
```

Room Configuration Response

The following XML schema is provided as a sample response after requesting a room configuration.

```
<RequestConfigurationResponse>
   <RequestID>152</RequestID>
   <RoomInformation>
      <ID>nnnnn</ID>
      <Name>Room 100</Name>
      <Location>North America, Washington, Redmond</Location>
      <Description></Description>
   </RoomInformation>
   <Capabilities>
      <PushModel Enabled=1 />
   </Capabilities>
   <CustomFields>
      <CustomField ID=”1234”>
         <CustomFieldName>Name</CustomFieldName>
         <CustomFieldType>Type</CustomFieldType>
         <CustomFieldValue>Value</CustomFieldValue>
      </Custom Field ID=”1234”>
   </CustomFields>
   <FusionVersion>MM.mm.rrrr.bb</FusionVersion>
</RequestConfigurationResponse>
```

NOTE: The syntax for the **FusionVersion** tag is major version (MM), minor version (mm), release (rrrr), and build (bb). For example, 13.01.rrrr.bb.

If there are no room custom properties defined in the Crestron Fusion database, then the `<CustomFields>` node will not be present.

Remove Schedule Request

Remove schedule requests are used to remove a schedule from a meeting room.

```
<RequestRemoveSchedule>
   <RequestID>1234</RequestID>
   <RoomID>nnnnnnnnn</RoomID>
   <MeetingEvents>
      <Event>
         <ID>555555</ID>
         <Recurring>True</Recurring>
         <NoShowCount>2</NoShowCount>
      </Event>
      <Event>
         <ID>6666666</ID>
         <Recurring>False</Recurring>
         <NoShowCount>0</NoShowCount>
      </Event>
   <Meeting Events>
   <InstanceEvents>
      <ID>1111</ID>
      <ID>2222</ID>
   </InstanceEvents>
</RequestRemoveSchedule>
```

Remove Schedule Response

The following XML schema is provided as a sample response after requesting to remove a room schedule.

```
<RemoveScheduleResponse>
   <RequestID>1234</RequestID>
   <RoomID>nnnnnnnnn</RoomID>
   <MeetingEvent>
      <ID>1111</ID>
      <Status>Success</Status>
      <FailedMsg></FailedMsg></Event>
   </MeetingEvent>
   <InstanceEvent>
      <ID>2222</ID>
      <Status>Failed</Status>
      <FailedMsg></FailedMsg>
   </InstanceEvent>
</RemoveScheduleResponse>
```

Create Schedule Request

Create schedule requests are used to create a new schedule for a meeting room.

```
<CreateSchedule>
   <RequestID>1234</RequestID>
   <RoomID>nnnnnnnnn</RoomID>
   <Event>
      <dtStart>2006-06-10T09:00:00Z</dtStart>
      <dtEnd>2006-06-10T09:45:00Z</dtEnd>
      <Organizer>John Doe</Organizer>
      <Subject>This is the subject</Subject
   </Event>
</CreateSchedule>
```

Create Schedule Response

The following XML schema is provided as a sample response after requesting to create a room schedule.

```
<CreateScheduleResponse>
   <RequestID>1234</RequestID>
   <RoomName></RoomName>
   <RoomID>nnnnnnnnn</RoomID>
   <MeetingID></MeetingID>
   <Errors>
      <Error></Error>
      <Error></Error>
   </Errors>
</CreateScheduleResponse>
```

Room Attribute List Request

The following XML schema are used to query for various room attribute functions.

Query to Change the State of an Attribute (Write)

```
<RequestRoomAttributeList>
   <RequestID>1234</RequestID>
   <Room>
      <RoomID>nnnnnnnnn</RoomID>
      <Write>
         <Attributes>
            <Attribute>
               <Join>d50</Join>
               <Value>1</Value>
            </Attribute>
         </Attributes>
      </Write>
   </Room>
</RequestRoomAttributeList>
```

Query to Obtain the State of an Attribute by Join (Read)

```
<RequestRoomAttributeList>
   <RequestID>1234</RequestID>
   <Room>
      <RoomID>nnnnnnnnn</RoomID>
      <Read>
         <Attributes>
            <Attribute>
               <Join>d50</Join>
            </Attribute>
         </Attributes>
      </Read>
   </Room>
</RequestRoomAttributeList>
```

Query to Obtain the State of an Attribute by Name (Read)

```
<RequestRoomAttributeList>
   <RequestID>1234</RequestID>
   <Room>
      <RoomID>nnnnnnnnn</RoomID>
      <Read>
         <Attributes>
            <Attribute>
               <Name>Hiya</Name>
            </Attribute>
         </Attributes>
      </Read>
   </Room>
</RequestRoomAttributeList>
```

Room Attribute List Response

The following XML schema is provided as a sample response after requesting a room attribute function.

```
<RoomAttributeListResponse>
   <RequestID>1234</RequestID>
   <Room>
      <RoomID>nnnnnnnnn</RoomID>
      <Attribute>
         <Name>SysPower</Name>
         <Join>d50</Join>
         <Value>True</Value>
         <IOMask>Read</IOMask>
      </Attribute>
      <Attribute>
         <Name>LampHours</Name>
         <Join>a25</Join>
         <Value>200</Value>
         <IOMask>Read/Write</IOMask>
      </Attribute>
   </Room>
</RoomAttributeListResponse>
```

Room Attribute Change Request

The following XML schema is used to make a change to a room attribute.

```
<RequestRoomAttributeChange>
   <RequestID>1234</RequestID>
   <RoomGUID>nnnnnnn</RoomGUID>
   <Join>a50</Join>
   <Value>True</Value>
</RequestRoomAttributeChange>
```

Room Local Time Request

The following XML schema is used to request the local time for a room in the system.

```
<LocalTimeRequest>
   <RequestID>1234</RequestID>
</LocalTimeRequest>
```

Room Local Time Response

The following XML schema is provided as a sample response after requesting the local time for a room in the system.

```
<LocalTimeResponse>
   <RequestID>1234</RequestID>
   <LocalDateTime>2019-11-05T13:15:30</LocalDateTime>
<LocalTimeResponse>
```

Help Request

Help requests are initiated by a device in the system and recorded in the Crestron Fusion database. Each new request begins a "session" that can be closed by either side (device or server).

Help Request (Initiation)

The following query is used to initiate a new help request.

```
<HelpRequest>
   <ID>room-unique-string</ID>
   <Message>text</Message
   <Severity>n</Severity>
   <Type>new_user</Type>
   <Organizer>Matt</Organizer>
</HelpRequest>
```

The security number is conventionally 1 to 3, but it can be any integer. The ID is a unique string for the room, such as a time stamp. The Type must be entered as shown above.

Help Response

The following query is used to respond to a help request.

```
<HelpRequest>
   <ID>room-unique-string</ID>
   <Type>new_user</Type>
   <RoomID>guid</RoomID>
   <Message>text</Message>
   <UserName>CrestronFusion-username</UserName>
</HelpRequest>
```

The help response is passed to the device as a result of a response from the web client. The ID must match the ID of the original request. Multiple responses are permitted.

Help Close

The following query is used to close a help session.

```
<HelpRequest>
   <ID>room-unique-string</ID>
   <Type>close</Type>
   <RoomID>guid</RoomID>
   <Message></Message>
   <UserName>CrestronFusion-username</UserName>
 </HelpRequest>
```

The web client user has the option to close the help session request. If the session is closed, the above XML will flow from the server to the device.

Help Cancel

The following query is used to cancel a help session.

```
<HelpRequest>
   <ID>room-unique-string</ID>
   <Type>cancel</Type>
   <Message>text</Message>
 </HelpRequest>
```

The device programming may allow a user to cancel an outstanding help request. The XML flows up to the server to clear the request from the web client.

Get Open Issues

The following query is used to get any open help requests from the server.

```
<HelpRequest>
   <Type>open_items</Type>
 </HelpRequest>
```

Get Open Issues Response

The following query is used to respond to a request to obtain open help requests.

```
<HelpRequest>
   <Type>open_items</Type>
   <OpenItems>
      <ID>20160204345621288</ID>
      [etc.]
   </OpenItems
</HelpRequest>
```

The <ID> tag is the identifier of the open help request. More than one <ID> tag may be present.

Room Schedule Request for Current/Next Meeting

The room schedule request is used to set the room schedule for the current or next meeting.

```
<RequestSchedule>
   <RequestID>FirstAppt55772</RequestID>
   <Start>2019-06-05T00:00:00</Start>
   <HourSpan>72</HourSpan>
   <Action ID="RequestFirstAppt"/>
   <FieldList>
      <Field>MeetingID</Field>
      <Field>InstanceID</Field>
      <Field>dtStart</Field>
      <Field>dtEnd</Field>
      <Field>Subject</Field>
      <Field>Organizer</Field>
      <Field>IsPrivate</Field>
      <Field>IsExchangePrivate</Field>
      [etc.]
   </FieldList>
</RequestSchedule>
```

The response is a single appointment in a `<ScheduleResponse>` tag.

Glossary

This section provides definitions for various terms that are used throughout this document.

- **Crestron Direct Connect Display**: This is a list of display manufacturers that are partnered with Crestron and that have implemented technology on their display chipsets to interface directly to Crestron Fusion without the need of a Crestron processor.
- **Fusion Room Device Definition**: This definition is the base programming module that is added as a device under the equipment view in SIMPL or as a room in Crestron Studio® software. All communication to the Crestron Fusion server (inbound and outbound) flows through this definition.
- **Fusion SSI Module:** This module is linked to all of the programming signals and is then tied directly to the Crestron Fusion Room Device definition. All other Crestron Fusion modules feed through this module to the Crestron Fusion Room Device definition to then send the information to the Crestron Fusion server (database).
- **Static Asset**: A static asset represents a physical device (either Crestron or third-party) in the system and is used in programming to provide status, control, and telemetry to Crestron Fusion.

Appendix A: Filtered Schedule Request Fields

Refer to the following table for the filtered schedule request fields.

Name	Returned Value	Example
Recurring	Whether the meeting is an instance of a recurring series	True
MeetingID	The GUID assigned to the meeting by Crestron Fusion	22403c59-8bc7-488d-a4f0-a5474faa6d75
RVMeetingID	(Synonym for **MeetingID**)	
dtStart	Starting date and time (local)	2019-03-21T13:00:00
dtEnd	Ending date and time (local)	2019-03-21T13:30:00
Organizer	Name and optional email address of the organizer of the meeting	"John Doe" <jdoe@yourcompany.com>
Subject	The meeting subject	Weekly Status Meeting
Body	The meeting body	Webex Meeting Call in Number: 555-555-5555
IsPrivate	Whether the meeting has been flagged as private in Crestron Fusion	True
IsExchangePrivate	Whether the meeting has been flagged as private in Microsoft® Exchange software or another groupware application	1
Attendees	The list of attendees	Refer to XML schema in Schedule Response (on page 65).

Crestron Electronics, Inc.
15 Volvo Drive, Rockleigh, NJ 07647
Tel: 888.CRESTRON
Fax: 201.767.7656
www.crestron.com

Reference Guide — 7898E
(2046497)
11/12/19
Specifications subject to
change without notice.