

EMBER+ CLIENT FOR CRESTRON

Revision: 1.00

Date: 07 October 2021



SUMMARY

Ultamation's suite of Ember+ client modules provide integration between a Crestron control system and an Ember+ provider. Specifically, the modules implement an ember+ "consumer", which allows Crestron systems to control, and receive feedback from, Ember+ enabled equipment. This suite does not offer ember+ "provider" functionality – e.g. this cannot be used to allow an ember+ control surface to control a Crestron system.

Ember+ has become a popular control protocol in the broadcast and professional Audio/Visual domain.

This datasheet provides the essential information for integration between an Ember+ provider and a Crestron program, determining the important Ember+ path information, and information on programming the module, with a host Crestron program.

This module is compatible with 3-Series/4-Series Crestron systems ONLY.

We recommend testing the module prior to purchase, to ensure the module meets your requirements, by downloading the module and using the 1 hour trial period. Refunds for licence keys issued cannot be given.

THE MODULE SUITE

There are two elements to the module suite:

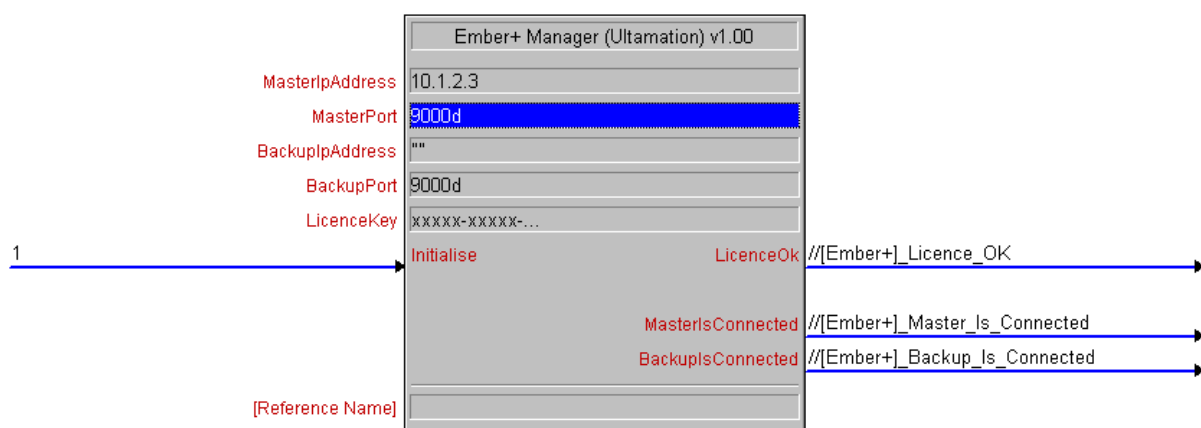
The Ember+ Manager

The first, and mandatory, module is the Ember+ Manager. There must be one, and only one, instance of this module in the program. The Ember+ Manager provides the connections to a primary and backup Ember+ provider. This provides support for Ember+ providers that support failover connections.

For each Ember+ endpoint, you must specify the IP address and port (default 9000) of the Ember+ provider device.

Finally, you must provide a licence key to enable continued use of the module outside of the 1 hour evaluation period. The licence key is tied to the processor MAC primary address, therefore, a deployment with a single Crestron processor will require a single licence; however, multiple Crestron processors will require an additional licence per additional processor.

All Ember+ modules must run in the same program slot as the Ember+ Manager.



The module parameters are as follows:

Type	Label	Description
P	MasterIpAddress	Enter the IP address of the main ember+ provider. This is mandatory.
P	MasterPort	Enter the IP port of the main ember+ provider. This is also mandatory and the default value is 9000.
P	BackupIpAddress	Enter a backup ember+ provider IP address if available. This must be a different ember+ provider from the master, or leave the parameter field as the empty string (""). The empty string tells the module that there is no backup, and the BackupPort parameter is then ignored.
P	BackupPort	As for MasterPort.
P	LicenceKey	Enter the licence key, as delivered in the purchase email for the processor on which the ember+ modules will be run.
I-D	Initialise	The master must be initialised before use. You can use a '1' on this input, or initialise following program start to deferred initialisation to spread load.
O-D	LicenceOk	Will go high IFF the licence key provided is validated for the processor. This will stay low if not, even though the 1 hour trial period will function. You can also check the processor's error log for licence validation information.
O-D	MasterIsConnected	Will go high when the primary ember+ provider connection is successfully made.
O-D	BackupsIsConnected	Will go high when the backup ember+ provider connection is successfully made.

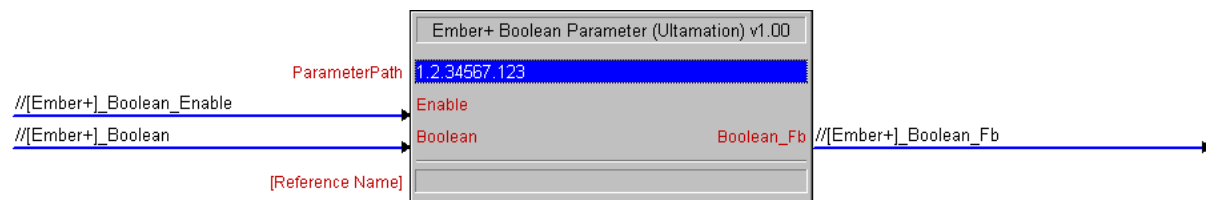
Where Type: P = Parameter, I/O = Input/Output and D/A/S = Digital/Analog/Serial

The Ember+ Parameter Modules

The rest of the module suite is made up of control modules. Each module represents a single instance of a particular parameter, function, or matrix.

Boolean Parameters

For Boolean/digital parameters, you should implement the Ember+ Boolean Parameter module.



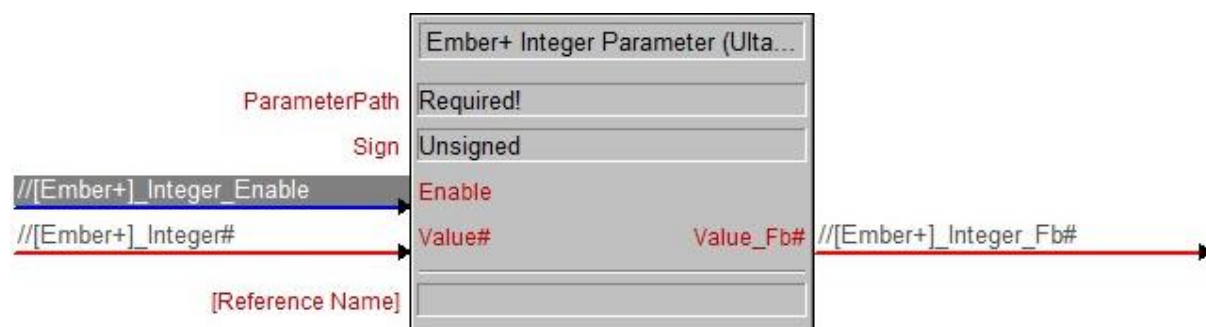
The critical element in each module is the ParameterPath. This identifies the parameter as it occurs in the "Ember+ tree" (see IMPLEMENTATION NOTES).

Type	Label	Description
P	ParameterPath	The "address" of the parameter to be controlled.
I-D	Enable	Pulse this signal to initialise the module – this should be done each time the ember+ manager is connected as it informs the ember+ provider to provide asynchronous feedback of the parameter's value.
I-D	Boolean	Set the ember+ value
O-D	Boolean_Fb	The current state of the ember+ value. The value is only valid following successful initialisation.

Where Type: P = Parameter, I/O = Input/Output and D/A/S = Digital/Analog/Serial

Integer Parameters

For integer parameters, the Ember+ Integer Parameter module is used.



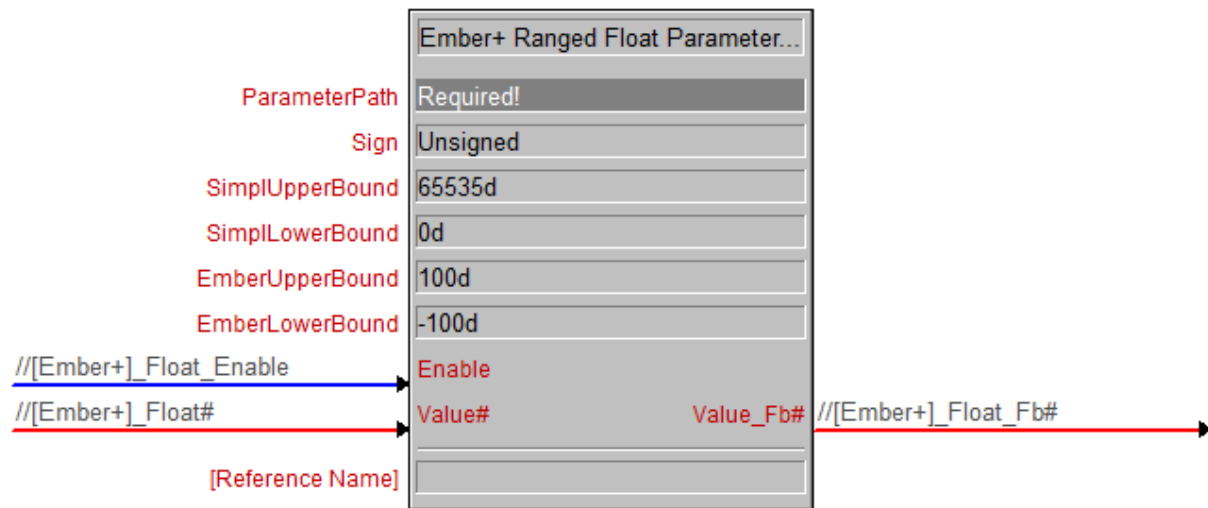
This provides support for integer values, signed and unsigned within the standard Crestron integer range.

Type	Label	Description
P	ParameterPath	The "address" of the parameter to be controlled.
P	Sign	Select whether the Crestron signal should be treated as signed or unsigned.
I-D	Enable	Pulse this signal to initialise the module – this should be done each time the ember+ manager is connected as it informs the ember+ provider to provide asynchronous feedback of the parameter's value.
I-A	Value#	Set the ember+ value
O-A	Value_Fb#	The current value of the ember+ parameter. The value is only valid following successful initialisation.

Where Type: P = Parameter, I/O = Input/Output and D/A/S = Digital/Analog/Serial

Real Parameters

With SIMPL being limited to 16-bit integer values, we have to make special allowances for real values in ember+. This is done through explicit scaling, and the module Ember+ Ranged Float.



The input value is passed scaled based upon the lower and upper bounds, specified in the parameters of the module. For example, if we wish to modify a parameter with a range between -100.00 and 100.00, the lower and upper bound would be set accordingly, and the input range would be set according to the available range within SIMPL; so – in this example – we'll assume the full 16-bit unsigned range 0-65535 (0-100%).

The table below shows a number of SIMPL input/output values and their corresponding ember+ values:

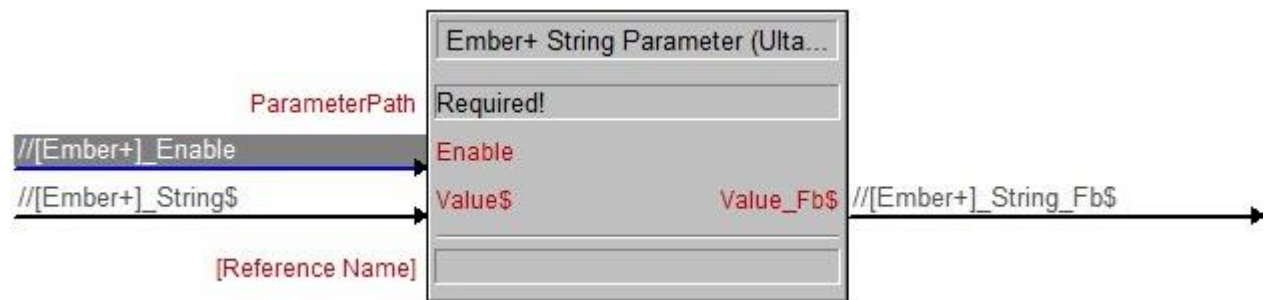
SIMPL	Ember+
0d / 0%	-100.00
65535d / 100%	100.00
20000d / 30.52%	-69.48
45000d / 68.67%	37.33

Type	Label	Description
P	ParameterPath	The "address" of the parameter to be controlled.
P	Sign	Select whether the Crestron signal should be treated as signed or unsigned.
P	SimplUpperBound	The SIMPL windows upper bound for the value
P	SimplLowerBound	The SIMPL windows lower bound for the value
P	EmberUpperBound	The Ember+ upper bound for the value
P	EmberLowerBound	The Ember+ lower bound for the value
I-D	Enable	Pulse this signal to initialise the module – this should be done each time the ember+ manager is connected as it informs the ember+ provider to provide asynchronous feedback of the parameter's value.
I-A	Value#	Set the ember+ value using the SIMPL range (converted internally to Ember+ range)
O-A	Value_Fb#	The current value of the ember+ parameter, scaled to the SIMPL range. The value is only valid following successful initialisation.

Where Type: P = Parameter, I/O = Input/Output and D/A/S = Digital/Analog/Serial

String Parameters

For text parameters, the Ember+ String Parameter module is used.

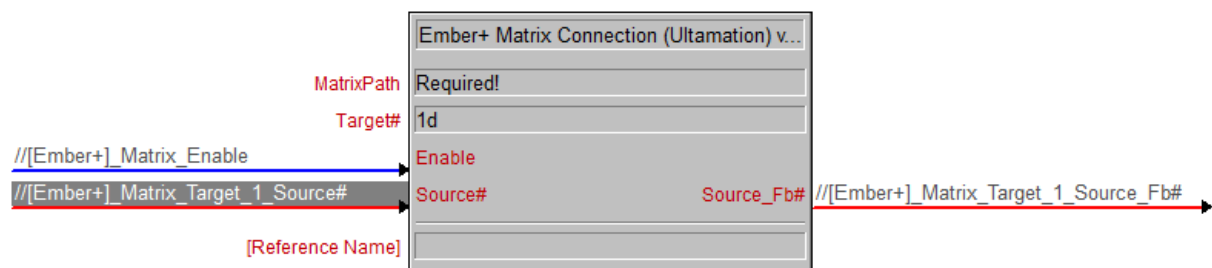


Type	Label	Description
P	ParameterPath	The "address" of the parameter to be controlled.
I-D	Enable	Pulse this signal to initialise the module – this should be done each time the ember+ manager is connected as it informs the ember+ provider to provide asynchronous feedback of the parameter's value.
I-S	Value\$	Set the ember+ value using the SIMPL serial signal.
O-S	Value_Fb\$	The current value of the ember+ parameter.

Where Type: P = Parameter, I/O = Input/Output and D/A/S = Digital/Analog/Serial

Matrices

Matrix outputs are controlled with one or more Ember+ Matrix Connection modules.



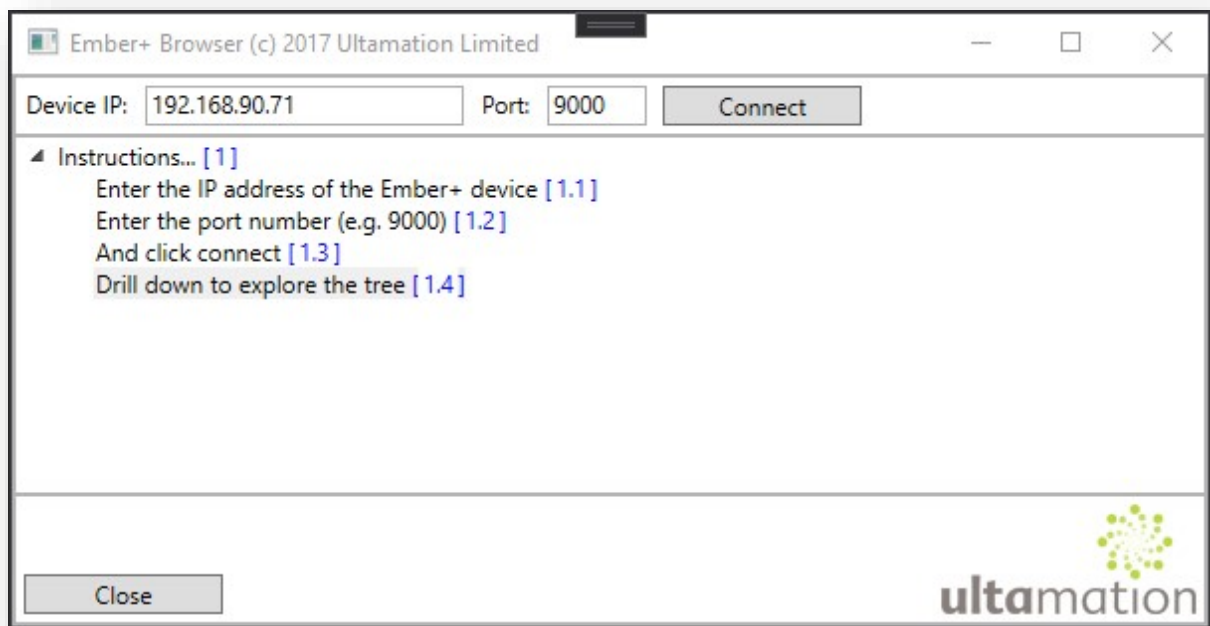
Type	Label	Description
P	MatrixPath	The "address" of the matrix to be controlled.
P	Target#	The output identifier, as defined in the Ember+ provider
I-D	Enable	Pulse this signal to initialise the module – this should be done each time the ember+ manager is connected as it informs the ember+ provider to provide asynchronous feedback of the matrix output's value.
I-A	Source#	Set the ember+ value for the desired source ID as defined in the Ember+ provider.
O-A	Source_Fb#	The current value of the target ID's current source.

Where Type: P = Parameter, I/O = Input/Output and D/A/S = Digital/Analog/Serial

Each output will require an instance of the module.

Please note – the Source and Target values may not map directly onto the hardware input and output indices. Please configure your system with a suitable mapping (e.g. In Riedel, this done in the Ember+ integration configuration).

IMPLEMENTATION NOTES



As previously described, each of the Ember+ modules requires a parameter of object path. This is the unique address of the object in the Ember+ tree and is device/manufacture specific. Some branches of a tree will be standard for a given manufacturer. For example, Riedel use standard values such as:

- Parameter paths ending in:
 - .330 - Gain
 - .331 - Phantom
 - .174 - Mute
 - .24581 - Pad
- The video matrix will likely be at address 1.2.0
 - and the subscription event will be on 1.2.0.3
- The audio matrix will likely be at address 1.2.1
 - And the subscription event will be on 1.2.1.3

Examples of actual path parameters are:

- XLR Phantom Power: 1.3.356714847.2.117445682.2.0.1.331
- Display Test Pattern: 1.3.589769984.2.33559557.2.0.1.176

To make determining the parameter paths simpler, we have developed a very simple tool which provides a way to browse the Ember+ object tree.

This tool is not required for run-time control of the Ember+ solution – it is only needed during commissioning.

To use the tool, open the Ember+ Browser, and follow the instructions as shown in the image above. You can copy the path from the tool directly into SIMPL Windows.

PROGRAMMING NOTES

Each of the module files (see below) should be placed either in the host program's project folder, or to make the Ember+ module suite available to all Crestron programs, in the SIMPL Windows installation's User SIMPL+ (for .usp, .ush and .clz files) directories. This pdf should be placed in the same directory for SIMPL's F1 help function to work properly.

The module consists of a number of SIMPL+ wrapper module to the core SIMPL# driver:

- EmberPlus Manager (Ultamation) v1.00.usp & .ush
- EmberPlus <module> Paramater (Ultamation) v1.00.usp & .ush
- EmberPlus Matrix Connection (Ultamation) v1.00.usp & .ush
- EmberPlusClient.clz
- Ember+ Integration Datasheet (Ultamation) v1.00.pdf (this file)

Multiple instances of the parameter modules can be used within a program, but only one instance of the Ember+ Manager module can be used in a program slot.

Support

For support questions, please contact Ultamation on support@ultamation.com.

Licence

This module (including software, images and any and all other associated assets distributed as part of the purchased download package) is licenced on a PER PROCESSOR basis.

A licence key is generated at the point of purchase and is linked at that time to specific information that MUST be provided at the time of purchase. A purchase should not be completed without correct information as refunds cannot be issued for errors or changes made to details following purchase.

The licence key for each processor will be delivered via email along with links to download the module. There is no physical delivery.

The module is provided without any warranty with respect to the Ember+ provider implementation. We will endeavour, through best efforts, to maintain the module's functionality and any bug fixes will be provided free-of-charge. Additional functionality may be released as a variation of this module and this will be a separate, purchasable, product.