Tyba TY-K-TURN Cresnet

Crestron module guide

The Tyba TY-K-TURN Cresnet module is used to interact with the TY-K-TURN when setup to use Cresnet mode (as opposed to IP). This module uses an instance of the C2N-CONNECTED-CHIP Cresnet symbol; please see the end of the document for more details.

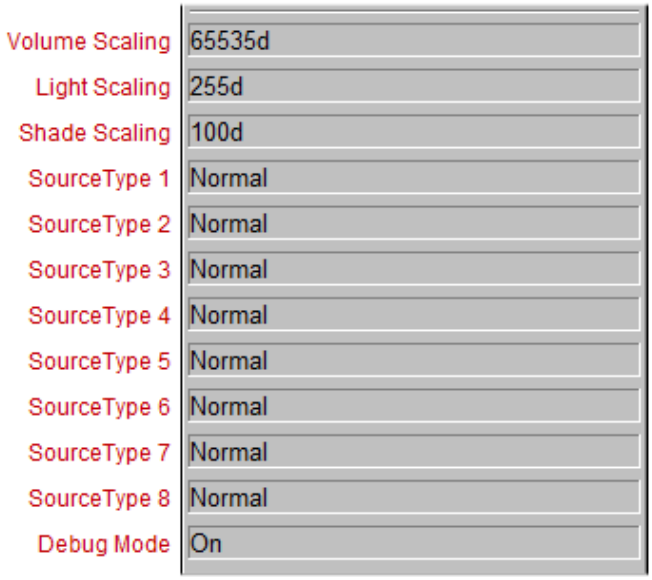
Required files

For the module to function correctly, the following files should be placed in either the project directory, or the central Crestron module directory:

Tyba TY-K-TURN Cresnet (version).umc
Tyba TY-K-TURN Cresnet (version).usp
Tyba TY-K-TURN Cresnet (version).ush

Usage – Parameters

When you drag in a new instance of the module, the first thing to do is to set the module parameters, as shown in the image below:



(please see next page for parameter descriptions)

Volume Scaling
This is an integer parameter that defines the maximum volume on the device (amplifier etc) that will be used to show volume feedback on the Turn. As every device will have a different maximum volume, this parameter is used to scale the volume into the correct range for the Turn.

Light Scaling
As with volume scaling above, this integer parameter denotes the maximum level used from the lighting system for light level feedback. KNX for example uses 0-255, whereas Crestron uses 0-65535.

Shade Scaling
As with volume/light scaling above, this integer parameter denotes the maximum level used from the shading system for shade level feedback. KNX for example uses 0-255, whereas Crestron uses 0-65535.

Source Type
Select Normal or Integrated. A normal source will update volume/media changes. Integrated will not update, as it will be updated by the integrated source (i.e. Sonos).

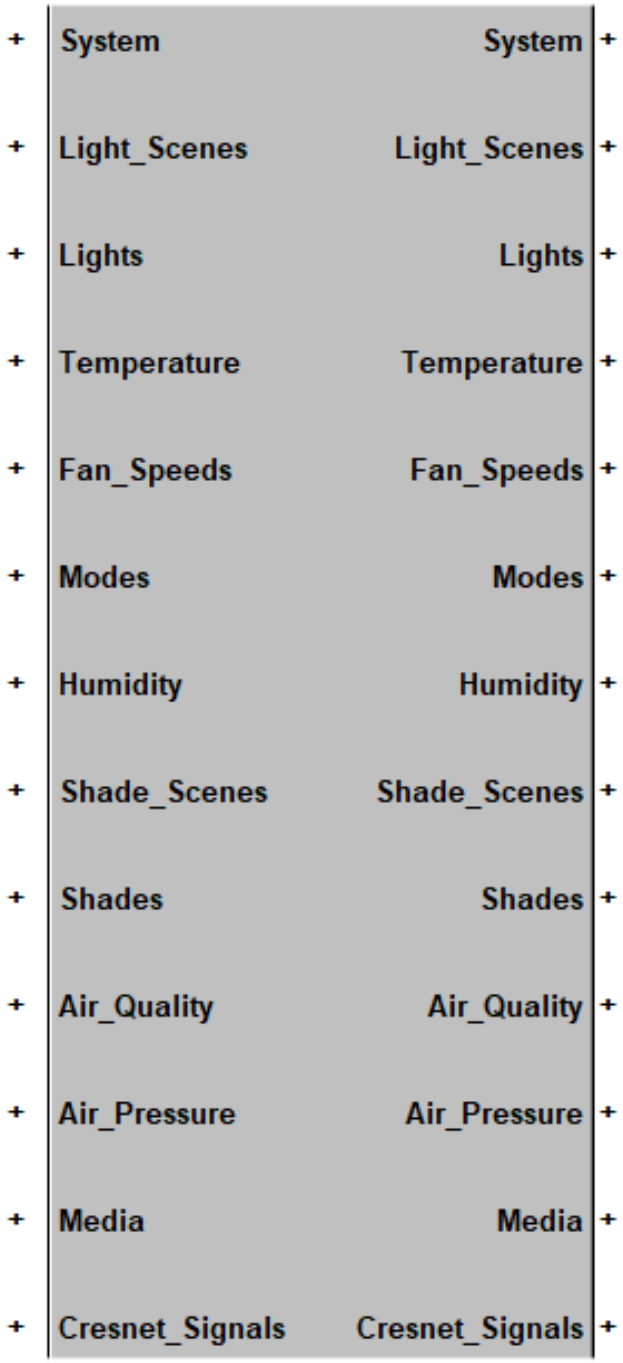
Debug Mode
This is a fixed list of 2 options, either On or Off. When set to On, the module will output additional debug information within Crestron Debugger, which makes it easier to find any potential control issues, such as IP disconnects etc.

(please see next page for module usage)

Usage – Signal Groups

The input/output signals on the module are broken down into sections, to match the available functions on the Turn. These are shown in the image below:

Each of these groups contains input/output signals, which are used to receive signals from the Turn, and send back feedback such as volume/active light scene etc.



These signals are a mix of digital/analog/serial and are described in more detail on the following pages.

System

The System signal group contains signals for initialising the module, outputting debug information, and discrete outputs from the temperature sensor, in Celsius and Fahrenheit. The discrete Celsius/Fahrenheit will always output their values, regardless of the settings in the config. This could be used when the Turn is simply a light controller, for example, to still provide a temperature reading for house management etc. Another example could be to use these for averaging across multiple sensors in a room, even in situations when the Turn is used for something simple, like controlling lights.

Inputs >>>

System_Initialise – this digital signal should be pulsed on processor bootup. It sets up the module ready for usage, based on the parameters etc. It is vital to pulse this signal on bootup, for the module to function properly.

>>> Outputs

System_Debug_Information$ - this serial signal can be used to trace debug information in Crestron Debugger. Whilst use of this signal is optional, it is recommended best practice to use it should any issues occur.

Discrete_Celsius – this analog signal will always output the Celsius reading of the internal temperature sensor, in hundreds; for example, 22.5 will output as 225.

Discrete_Fahrenheit – this analog signal will always output the Fahrenheit reading of the internal temperature sensor, in hundreds; for example, 66.5 will output as 665.

Discrete_Celsius$ – this serial signal will always output the Celsius reading of the internal temperature sensor as a floating point string, for example “22.5”

Discrete_Fahrenheit$ – this serial signal will always output the Fahrenheit reading of the internal temperature sensor, as a floating point string, for example “66.5”

(please see next page for continued signal groups)

Light_Scenes

The Light_Scenes signal group contains signals for selecting scenes, as well as signals for incrementing/decrementing the current scene, and to show whether scenes are currently on/off, including feedback signals for all.

Module Inputs >>>

Light_Scene_xx_FB (*where xx = Off to 7)* – these signals should be latched to display the currently selected light scene on the Turn. Only 1 of these should be latched at any one time. These signals should be tied to the scene feedback from the lighting system.

Light_Scene_Switched_On_FB/Light_Scene_Switched_Off_FB – when not using scenes, these 2 inputs can be used as an absolute on/off feedback to the Turn. Only one of these signals should be high at any one time.

Light_Scene_Increment/Light_Scene_Decrement– these 2 inputs provide an option for selecting scenes on the Turn. Pulsing increment will move up the scenes by 1 position (scene_1 -> scene_2 -> scene_3). Pulsing decrement will move down the scenes by 1 position (scene_3 -> scene_2 -> scene_1).

>>> Module Outputs

Light_Scene_xx (*where xx = Off to 7)* – these signals will pulse as the user selects light scenes on the Turn. These should be tied to the corresponding scene signals on the lighting system.

Light_Scene_Switch_On/Light_Scene_Switch_Off – as an alternative to scenes, these 2 outputs provide an absolute on/off from the Turn. If the user turns the outer wheel all the way to the left, the off will be pulsed. If the user turns the wheel to the right, the on signal will pulse.

(please see next page for continued signal groups)

Lights

The Lights signal group contains signals for individual lighting circuits, which include an on/off for each circuit, as well as a light level, all with feedback.

Module Inputs >>>

Light_xx_Level_FB (*where xx = 1 to 5)* – these analog signals can be used to display the actual light level for each of the 5 circuits. The scaling used on this input relates to the parameters described on page 2. These signals should be tied to the relevant level signal from the lighting system.

Light_xx_On_FB/Light_xx_Off_FB (*where xx = 1 to 5)* – when not using levels, these 2 inputs can be used to show if a light circuit is currently on or off. Only one of these signals should be high at any one time.

>>> Module Outputs

Light_xx_Level (*where xx = 1 to 5)* – these analog signals can be used to control the level of a lighting circuit. They output a value as the user turns the outer wheel on the Turn with a lights page selected (up to 5). The range of these signals will be determined by the Light Scaling parameter as described on page 2. These signals should be tied to the relevant light level signal on the lighting system.

Light_xx_On/Light_xx_Off (*where xx = 1 to 5)* – these digital signals can be used as an absolute on/off when not using levels. If the user turns the outer wheel on the Turn all the way to 0% (with 1 of 5 lights selected), then the off signal will pulse. If the user turns to any value higher than 0%, then the on signal will pulse instead.

(please see next page for continued signal groups)

Shade_Scenes

The Shade_Scenes signal group contains signals for selecting shade scenes, as well as signals for incrementing/decrementing the current scene.

Module Inputs >>>

Shade_Scene_xx_FB (*where xx = 1 to 8)* – these signals should be latched to display the currently selected shade scene on the Turn. Only 1 of these should be latched at any one time.

Shade_Scene_On_FB/Shade_Scene_Off_FB – when not using scenes, these 2 inputs can be used as an absolute on/off feedback to the Turn. Only one of these signals should be high at any one time.

Shade_Scene_Increment/Shade_Scene_Decrement– these 2 inputs provide an option for selecting shade scenes on the Turn. Pulsing increment will move up the shade scene by 1 position (scene_1 -> scene_2). Pulsing decrement will move down the shade scene by 1 position (scene_2 -> scene_1).

>>> Module Outputs

Shade_Scene_xx (*where xx = 1 to 8)* – these signals will pulse as the user selects shade scenes on the Turn. These should be tied to logic that sets the various shades to pre-defined levels.

Shade_Scene_On/Shade_Scene_Off – as an alternative to scenes, these 2 outputs provide an absolute on/off from the Turn. If the user turns the outer wheel all the way to the left, the off will be pulsed. If the user turns the wheel to the right, the on signal will pulse.

(please see next page for continued signal groups)

Shades

The Shade signal group contains signals for individual shades (curtains/blinds etc), which include 2 options for control; either an analog "position" signal (for use on systems that allow for absolute blind positions), or digital open/close/stop signals.

Module Inputs >>>

Shade_xx_Position_FB (*where xx = 1 to 5)* – these analog signals can be used to display the actual position of each shade on the Turn. The scaling used on these inputs relates to the Shade Level parameter described on page 2. These signals should be tied to the relevant position signal from the shading system.

Shade_xx_Open_FB (*where xx = 1 to 5)* – on systems that do not provide a position signal for each shade, these digital signals can be used to display on the Turn that shades are open.

Shade_xx_Close_FB (*where xx = 1 to 5)* – on systems that do not provide a position signal for each shade, these digital signals can be used to display on the Turn that shades are closed.

>>> Module Outputs

Shade_xx_Position (*where xx = 1 to 5)* – these analog signals can be used to control the position of a shade. They output a value as the user turns the outer wheel on the Turn with a shade page selected (up to 5). The range of these signals will be determined by the Shade Scaling parameter as described on page 2. These signals should be tied to the relevant shade position signal on the shading system.

Shade_xx_Open (*where xx = 1 to 5)* – on systems that do not provide a position signal for each shade, these digital signals can be used to perform an open function on the shade. This will pulse high if the user scrolls the outer wheel on the Turn to the **left**, with a shade selected.

Shade_xx_Close (*where xx = 1 to 5)* – on systems that do not provide a position signal for each shade, these digital signals can be used to perform a close function on the shade. This will pulse high if the user scrolls the outer wheel on the Turn to the **right**, with a shade selected.

(please see next page for continued signal groups)

Temperature

The Temperature signal group contains signals for displaying the current room temperature and setpoint, as well as outputs for changing the setpoint from the Turn. Both analog and serial signals are available, depending on the HVAC system being used.

Module Inputs >>>

Temperature_Setpoint_FB – this analog signal can be used to display the actual temperature setpoint of the room, on the Turn. As Crestron does not use floating point numbers, this number should be in the hundreds range, for example to display 22.5 the value should be 225. This signal should be tied to the temperature setpoint analog feedback from the HVAC system.

Temperature_Setpoint_FB$ – this serial signal can be used as an alternative to the analog signal above, to display the actual temperature setpoint of the room, on the Turn. This should be in the xx.x format (for example 22.5). This signal should be tied to the setpoint serial feedback from the HVAC system.

Temperature_Current_FB – this analog signal can be used to display the actual current temperature of the room, on the Turn. As Crestron does not use floating point numbers, this number should be in the hundreds range, for example to display 22.5 the value should be 225. This signal should be tied to the temperature analog feedback from the HVAC system.

Temperature_Current_FB$ – this serial signal can be used as an alternative to the analog signal above, to display the actual current temperature of the room, on the Turn. This should be in the xx.x format (for example 22.5). This signal should be tied to the temperature serial feedback from the HVAC system.

Temperature_Setpoint_Up – this digital signal can be used to increment the setpoint on the turn by 0.5, on systems that do not provide an output for analog/serial.

(please see next page for continued temperature signals)

Temperature_Setpoint_Down – this digital signal can be used to decrement the setpoint on the turn by 0.5, on systems that do not provide an output for analog/serial.

>>> Module Outputs

Temperature_Setpoint – this analog signal can be used to change the setpoint for the room on the HVAC system. When the user is on the Temperature page on the Turn, this signal will output a value as the user turns the outer wheel. As Crestron does not use floating point numbers, this number will be in the hundreds range, for example if the Turn is displaying 22.5 the value output will be 225. This signal should be tied to the analog temperature setpoint signal on the HVAC system.

Temperature_Setpoint_FB$ – this serial signal can be used as an alternative to the analog signal above, to change the setpoint of the room on the HVAC system. This will be in the xx.x format (for example 22.5). This signal should be tied to the serial setpoint signal on the HVAC system.

Temperature_Current – this analog signal can be used to display the temperature reading from the internal sensor, when the config is set to use internal temperature. This will be in the selected format within the config, either Celsius or Fahrenheit. As Crestron does not use floating point numbers, this number will be in the hundreds range, for example 22.5(c) will be 225, or 66.5(f) will be 665. This signal will only output when the temperature page is used on the Turn. If you need internal temperature readings without using a temperature page, please refer to the discrete temperatures under the system section.

Temperature_Current$ – this serial signal can be used to display the temperature reading from the internal sensor, when the config is set to use internal temperature. This will be in the selected format within the config, either Celsius or Fahrenheit. This will be in the xx.x format, for example 22.5(c) or 66.5(f). This signal will only output when the temperature page is used on the Turn. If you need internal temperature readings without using a temperature page, please refer to the discrete temperatures under the system section.

(please see next page for continued signal groups)

Humidity

The Humidity signal group contains signals for displaying the current room humidity and setpoint, as well as outputs for changing the setpoint from the Turn. Both analog and serial signals are available, depending on the HVAC system being used.

Module Inputs >>>

Humidity_Setpoint_FB – this analog signal can be used to display the actual humidity setpoint of the room, on the Turn. As Crestron does not use floating point numbers, this number should be in the hundreds range, for example to display 22% the value should be 220. This signal should be tied to the humidity setpoint analog feedback from the HVAC system.

Humidity_Setpoint_FB$ – this serial signal can be used as an alternative to the analog signal above, to display the actual humidity setpoint of the room, on the Turn. This should be in the xx format (for example 22). This signal should be tied to the humidity setpoint serial feedback from the HVAC system.

Humidity_Setpoint_Up – this digital signal can be used to increment the humidity setpoint on the turn by 1%, on systems that do not provide an output for analog/serial.

Humidity_Setpoint_Down – this digital signal can be used to decrement the humidity setpoint on the turn by 1%, on systems that do not provide an output for analog/serial.

>>> Module Outputs

Humidity_Setpoint – this analog signal can be used to change the humidity setpoint for the room on the HVAC system. When the user is on the Humidity page on the Turn, this signal will output a value as the user turns the outer wheel. As Crestron does not use floating point numbers, this number will be in the hundreds range, for example if the Turn is displaying 22% the value output will be 220. This signal should be tied to the analog humidity setpoint signal on the HVAC system.

(please see next page for continued humidity signals)

Humidity_Setpoint_FB$ – this serial signal can be used as an alternative to the analog signal above, to change the humidity setpoint of the room on the HVAC system. This will be in the xx.x format (for example 22.5). When the user is on the Humidity page on the Turn, this signal will output a value as the user turns the outer wheel. This signal should be tied to the serial humidity setpoint signal on the HVAC system.

Fan Speeds

The Fan Speeds signal group contains signals for selecting fan speeds, as well assignals for incrementing/decrementing the current speed, and to show whether fans are currently on/off, including feedback signals for all.

Module Inputs >>>

Fans_Off_FB – this digital signal should be latched to display that fans are off on the turn. This signal should be tied to the fans off feedback from the HVAC system.

Fans_xx_FB (*where xx = 1 to 7)* – these signals should be latched to display the currently selected fan speed on the Turn. Only 1 of these should be latched at any one time, unless Fans_Off_FB is latched, in which case none of these should be. These signals should be tied to the fan speed feedback from the HVAC system.

Fans_Switched_On_FB/Fans_Switched_Off_FB – these 2 signals can be used as an absolute fans on/off feedback to the Turn when individual speeds are not being used. Only one of these should be high at any one time.

Fans_Increment/Fans_Decrement– these 2 inputs provide an alternative option for changing fan speeds on the Turn. Pulsing these will increment/decrement the fanspeed on the Turn by 1 speed.

>>> Module Outputs

Fans_Off – this digital signal will pulse when the user selects off on the Turn when on the fans page. Should be tied to the fans off signal in the HVAC system.

(please see next page for continued fan signals)

Fans_xx (*where xx = 1 to 7)* – these signals will pulse as the user selects fan speeds on the Turn. These should be tied to the corresponding fan speed signals on the HVAC system.

Fans_Switch_On/Fans_Switch_Off – these 2 inputs provide an absolute on/off for the fan speeds from the Turn. If the user moves the outer wheel all the way to the left, the off signal will pulse. If the user moves the outer wheel to the right, the on signal will pulse.

Modes

The Modes signal group contains signals for selecting HVAC modes, as well as signals for incrementing/decrementing the current mode, including feedback signals for all.

Module Inputs >>>

Modes_xx_FB (*where xx = 1 to 8)* – these signals should be latched to display the currently selected HVAC mode on the Turn. Only 1 of these should be latched at any one time. These signals should be tied to the HVAC modes feedback from the HVAC system.

Modes_Switched_On_FB/Modes_Switched_Off_FB – these 2 signals can be used as an absolute modes on/off feedback to the Turn when individual modes are not being used. Only one of these should be high at any one time.

Modes_Increment/Modes_Decrement– these 2 inputs provide an alternative option for changing modes on the Turn. Pulsing these will increment/decrement the mode on the Turn by 1.

>>> Module Outputs

Modes_Off – this digital signal will pulse when the user selects the lowest mode on the Turn when on the modes page. Should be tied to the relevant mode signal in the HVAC system.

Modes_xx (*where xx = 1 to 8)* – these signals will pulse as the user selects HVAC modes on the Turn. These should be tied to the corresponding mode signals on the HVAC system.

(please see next page for continued mode signals)

Modes_Switch_On/Modes_Switch_Off – these 2 inputs provide an absolute on/off for the modes from the Turn. If the user rotates the outer wheel all the way to the left, the off signal will pulse. If the user rotates the outer wheel to the right, the on signal will pulse.

Media

The Media signal group contains signals for showing play/pause status of the currently selected source, showing whether the audio is currently muted or not, volume level and increment, selecting sources and system off, and displaying source artwork and metadata.

Module Inputs >>>

Play/Pause_FB – this digital signal should be buffered from the currently selected source (where available), and will change the play/pause icon on the Turn depending on its state; high will show the playing icon , low will show the pause icon.

Mute_FB – this digital signal should be held high when the audio is muted, to show the mute on icon on the Turn. When low the un-muted signal will be displayed on the Turn.

Prev/Next_FB – these digital signals can be buffered from the currently selected source (where available) but are optional.

Volume_Increment/ Volume_Decrement– these 2 inputs can be used when direct volume level feedback is not available. Pulsing these will increment/decrement the volume on the Turn by 1%.

Volume_FB – this analog signal can be used to display the actual volume in the room and uses the Volume Scaling parameter as shown on page 2. This should be tied to the volume feedback signal from the amplifier (etc).

System_Off_FB – this digital input can be used to set the AV page to an offstate, clearing any artwork and metadata.

(please see next page for continued Media signals)

Select_Source_xx_FB (*where xx = 1 to 8)* – these digital inputs can be used to select one of the 8 sources on the AV page.

Source_Artwork_FB$ – this serial input can be used to display artwork on the AV page, by sending in the URL of an image.

Source_Metadata_1_FB$ – this serial input can be used to display a line of meta-data on the first line of the AV page, such as playing track name.

Source_Metadata_2_FB$ – this serial input can be used to display a line of meta-data on the second line of the AV page, such as playing artist name.

>>> Module Outputs

Play/Pause – this digital signal can either be tied directly to the Play/Pause signal of the currently selected device, or where available, should be buffered to the direct Play and Pause, depending on the Play/Pause state of the source.

Prev – this digital signal will pulse as the user presses the previous button on the Turn display whilst on the Media page. Should be tied to the prev/rew (etc) signal ofthe currently selected source.

Next – this digital signal will pulse as the user presses the next button on the Turn display whilst on the Media page. Should be tied to the next/ffwd (etc) signal ofthe currently selected source.

Volume – this analog signal will output a value as the user moves the outer wheel on the Turn when on the Media page. This will be scaled according to the Volume Scaling parameter as shown on page 2. This should be tied directly tothe volume level input of the rooms amplifier (etc).

System_Off – this digital output will pulse when the user presses the system off button on the Media page.

Select_Source_xx (*where xx = 1 to 8)* – these digital inputs will pulse when the user selects one of the 8 sources on the Media page.

(please see next page for continued signal groups)

Air Quality/Pressure

The air quality/pressure signal groups contain signals for showing air quality/pressure on the Turn; this can either be a level (analog), a message (serial).

>>> Module Inputs

Air_Quality/Pressure_FB – these analog signals can be used to display an info level on the Turn. Air quality, displayed as ppm, air pressure is displayed as pa.

Air_Quality/Pressure_FB$ – these serial signals can be used to display an info level on the Turn. Air quality, displayed as ppm, air pressure is displayed as pa.

Please see the following pages for Cresnet information.

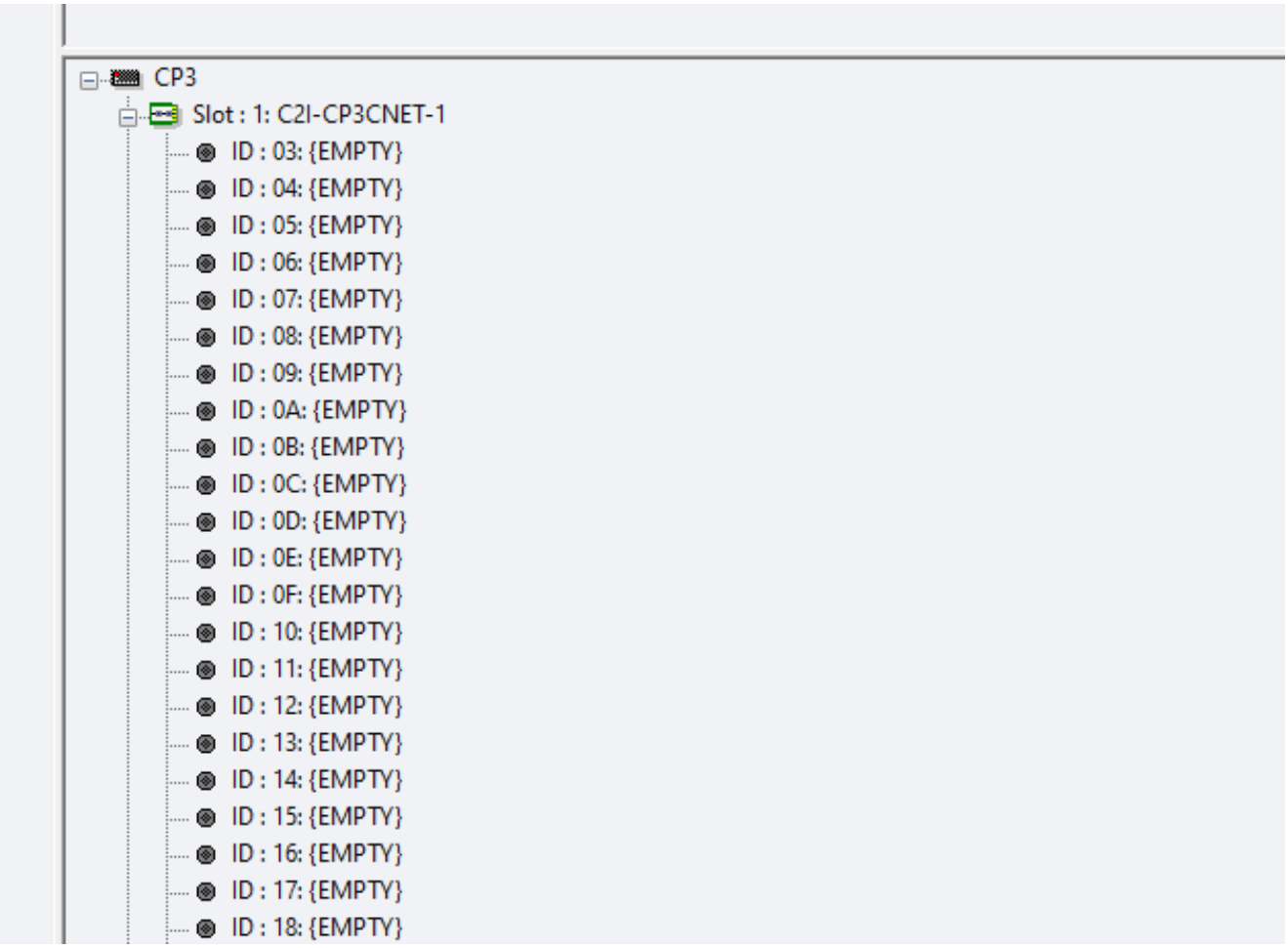
Cresnet

The Cresnet signal group contains all the digital/serial joins that need to be tied to the C2N-CONNECTED-CHIP. These joins are numbered and labelled input/output for ease of use. The joins simply need to match the numbers on the Cresnet device and be placed on either the input (press) or output (fb).

Cresnet and the C2N-CONNECTED-CHIP

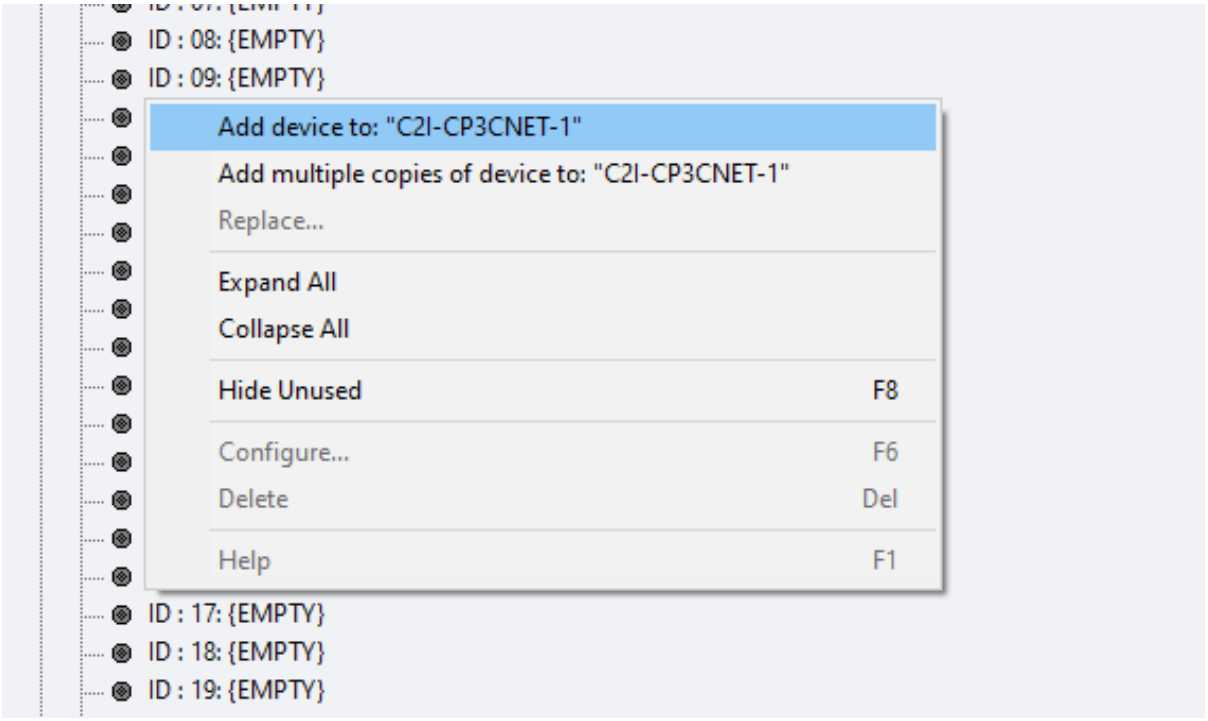
When the Turn is set to communicate over Cresnet, the module needs a way to send/receive signals via the Cresnet bus. This is achieved using a C2N-CONNECTED-CHIP, which is included in the Crestron device database.

After setting/determining the Cresnet ID of the Turn, a new device needs to be added using the hardware config in SIMPL:



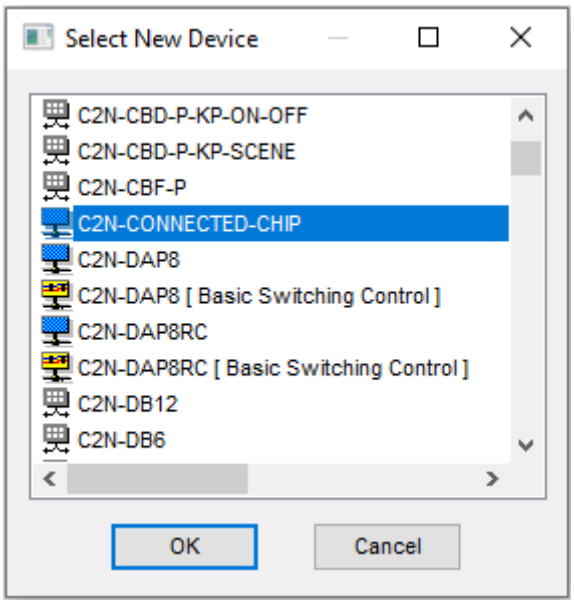
The configure section on a CP3, with the Cresnet (CNET) devices expanded. Currently no devices added.

For this example, we will assume that out Turn is set to communicate on Cresnet ID **0A**. The easiest way to add a C2N-CONNECTED-CHIP is to right-click the relevant ID, and choose “Add device to”:



After right-clicking ID 0A, select “Add device to” at the top of the popup options.

After selecting add device, we can then simply scroll down to C2N-CONNECTED-CHIP, and double-click to add on the selected ID (0A):



The C2N-CONNECTED-CHIP can be found in the “Select New Device” list

Cresnet and the C2N-CONNECTED-CHIP (continued)

With the C2N-CONNECTED-CHIP added to the program we can then tie the module Cresnet signals to the device.

-	Cresnet_Signals	Cresnet_Signals	-
CresnetTurn_Digital_In_1	[Cresnet_Digital_In_1]	resnet_Digital_Out_11_FB]	CresnetTurn_Digital_Out_11_FB
CresnetTurn_Digital_In_11	[Cresnet_Digital_In_11]	resnet_Digital_Out_15_FB]	CresnetTurn_Digital_Out_15_FB
CresnetTurn_Digital_In_35	[Cresnet_Digital_In_35]	resnet_Digital_Out_19_FB]	CresnetTurn_Digital_Out_19_FB
CresnetTurn_Digital_In_53	[Cresnet_Digital_In_53]	resnet_Digital_Out_23_FB]	CresnetTurn_Digital_Out_23_FB
CresnetTurn_Digital_In_56	[Cresnet_Digital_In_56]	resnet_Digital_Out_27_FB]	CresnetTurn_Digital_Out_27_FB
CresnetTurn_Digital_In_58	[Cresnet_Digital_In_58]	resnet_Digital_Out_31_FB]	CresnetTurn_Digital_Out_31_FB
CresnetTurn_Digital_In_62	[Cresnet_Digital_In_62]	resnet_Digital_Out_35_FB]	CresnetTurn_Digital_Out_35_FB
CresnetTurn_Digital_In_66	[Cresnet_Digital_In_66]	resnet_Digital_Out_39_FB]	CresnetTurn_Digital_Out_39_FB
CresnetTurn_Digital_In_70	[Cresnet_Digital_In_70]	resnet_Digital_Out_43_FB]	CresnetTurn_Digital_Out_43_FB
CresnetTurn_Digital_In_74	[Cresnet_Digital_In_74]	resnet_Digital_Out_47_FB]	CresnetTurn_Digital_Out_47_FB
CresnetTurn_Digital_In_111	[Cresnet_Digital_In_111]	resnet_Digital_Out_53_FB]	CresnetTurn_Digital_Out_53_FB
		[Cresnet_Digital_Out_56_FB]	CresnetTurn_Digital_Out_56_FB
		[Cresnet_Digital_Out_58_FB]	CresnetTurn_Digital_Out_58_FB
		[Cresnet_Digital_Out_62_FB]	CresnetTurn_Digital_Out_62_FB
		[Cresnet_Digital_Out_66_FB]	CresnetTurn_Digital_Out_66_FB
		[Cresnet_Digital_Out_70_FB]	CresnetTurn_Digital_Out_70_FB
		[Cresnet_Digital_Out_74_FB]	CresnetTurn_Digital_Out_74_FB
		[Cresnet_Digital_Out_111_FB]	CresnetTurn_Digital_Out_111_FB
Cresnet_Serial_In_10$	[Cresnet_Serial_In_10$]	[Cresnet_Serial_Out_10$]	Cresnet_Serial_Out_10$
Cresnet_Serial_In_14$	[Cresnet_Serial_In_14$]	[Cresnet_Serial_Out_12$]	Cresnet_Serial_Out_12$
Cresnet_Serial_In_18$	[Cresnet_Serial_In_18$]	[Cresnet_Serial_Out_13$]	Cresnet_Serial_Out_13$
Cresnet_Serial_In_22$	[Cresnet_Serial_In_22$]	[Cresnet_Serial_Out_14$]	Cresnet_Serial_Out_14$

The Cresnet signal group on the module, expanded to show the digital/serial joins that need to be tied to the C2N-CONNECTD-CHIP.

It is simply a case of tying the “In” signals to the incoming joins on the C2N-CONNECTED-CHIP, and the “Out” signals to the outgoing joins. The numbers need to match the join numbers on the device, for proper operation; so “Serial_Out_10$” will obviously need to be tied to the serial output signal number 10 on the device.

Although not essential, it is best practice to tie ALL the Cresnet joins to the C2N-CONNECTED-CHIP. This will ensure the module functions as expected.

Only digital and serial joins are used, there are no analog signals to tie to the Cresnet.