

BUTTONS

Revision	0.43
Date	17/02/2025

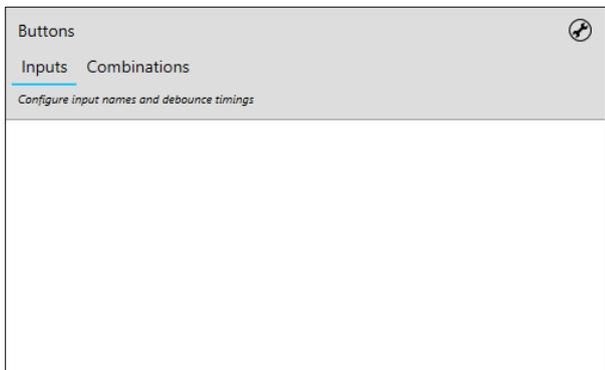
Buttons overview

Configuring buttons within Toolset is one of the best ways to expand functionality and allow flexibility when you create a setup. The following shows how you can configure buttons (with an emphasis on CCW & CSB), how buttons within Toolset are structured, and some examples of how to use them.

Configure button inputs

Add buttons

By default, on the **Buttons** node there are no entries on the **Inputs** page. Unlike a Maths channel or Logic channel, which are software defined, buttons are hardware defined. On devices such as the CCW and the CSB which communicate via CAN, the button properties are imported from the associated device CAN.



When you configure a CAN device such as the CCW or CSB, it comes with an associated CAN stream to transmit data between the primary device and the CAN device. Once the device stream has been imported to Toolset (see **Streams**), it automatically updates the **Buttons** node to show the available buttons.

You can also generate your own virtual buttons via CAN streams and define any received channels as a 'Button Group'.

When the device stream is imported (a CCW stream in the example below), button inputs are shown on the **Buttons** node in the format of 'Name (Source)'.

Buttons	
Inputs	Combinations
CCW Switch1 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch2 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch3 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch4 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch5 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch6 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch7 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch8 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch9 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch10 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch11 Shift L (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch12 Shift R (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch13 AD L (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch14 AD R (Stat_Switches_CCW) <i>debounced at source</i>

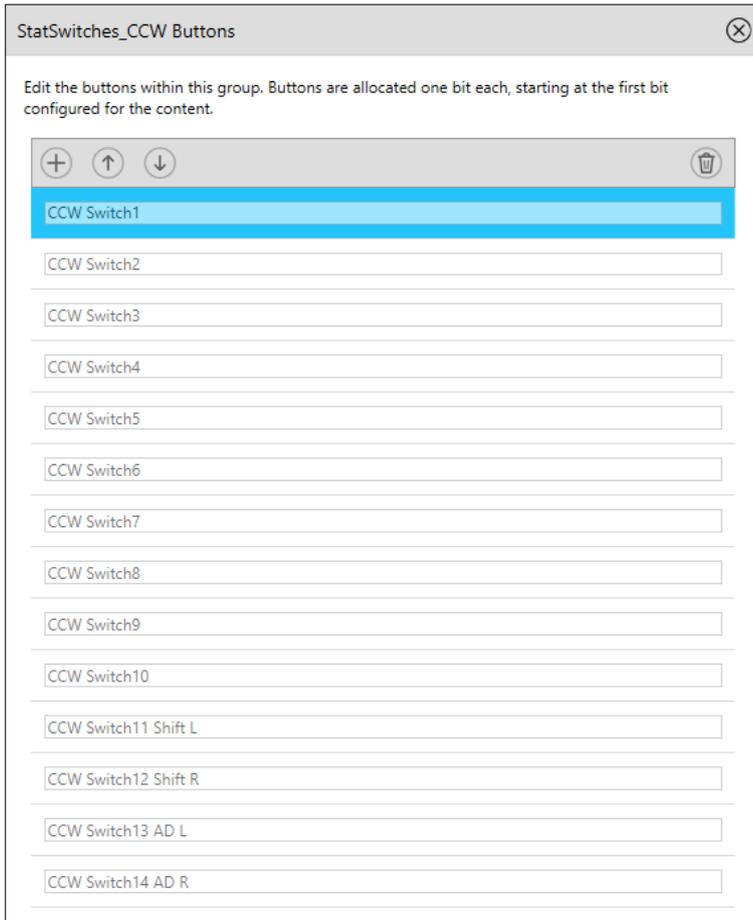
Name buttons

The name of the button is configured in the **CAN Streams** node. To edit the name of the button, select the 'Button Group' packet content (1), and then click **Edit Buttons** (2).

The screenshot shows the 'CCW_MK2_Decode_1.2 (CAN Stream Decode)' configuration window. On the left, a 'Packets' list shows 'Board Status' selected. The main area is divided into 'General' and 'Content' sections. The 'Content' section contains a table of packet contents:

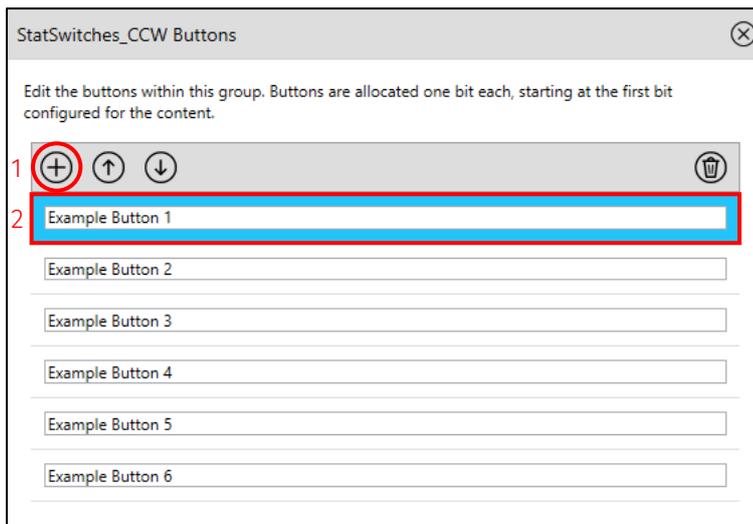
Name	Type	Start Bit	Length
Stat_Switches_CCW	Button Group	0	22
Stat_CAN_Term_CCW	Channel	23	1
Temp_Box_CCW	Channel	24	8
V_Battery_CCW	Channel	32	8
Stat_Device_ID_CCW	Channel	40	4
Stat_Software_Ver_CCW	Channel	44	4
Stat_CVW_Switches_CCW	Channel	48	16

The 'Stat_Switches_CCW' row is highlighted in blue and marked with a red box labeled '1'. To the right of this row, a configuration panel is visible with a button labeled 'Edit Buttons...' also highlighted with a red box labeled '2'.



If a pre-compiled CAN stream for a device (that has been locked) is imported, you cannot edit the default button name, and it is greyed-out.

If a new CAN stream is compiled, you can add buttons by clicking the + icon (1), and then enter the button name (2).



Configure button properties

From the **Buttons** node, the **Button Configuration** option is displayed. Here the name of the button is displayed, and you can define the **Timing configuration**. Select or unselect the **Use default timings** option (1).



General configuration

Name

Timing configuration

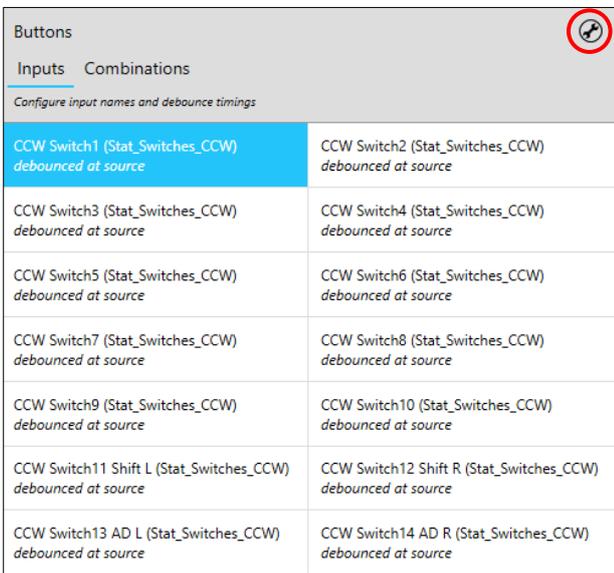
Use default timings

Hold time s

Long hold time s

Click the 'Wrench' icon on the **Buttons** node to configure default timings.

Click the 'wrench' icon to display a dialog box where you can configure the default timing configuration.



Buttons	
Inputs	Combinations
Configure input names and debounce timings	
CCW Switch1 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch2 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch3 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch4 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch5 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch6 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch7 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch8 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch9 (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch10 (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch11 Shift L (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch12 Shift R (Stat_Switches_CCW) <i>debounced at source</i>
CCW Switch13 AD L (Stat_Switches_CCW) <i>debounced at source</i>	CCW Switch14 AD R (Stat_Switches_CCW) <i>debounced at source</i>

You can configure timing configuration settings such as 'On', 'On Hold', 'Off', and 'Ignore' times under **Debounce**. 'Click', 'Hold', and 'Long Hold' timings are configurable under the **Timings** section.



The default configuration applied to all items.

Note that:

- An individual item may override the default configuration.
- In addition, some items may not support some or all of the settings provided here, in which case the settings provided here will not apply.

Debouncing	Timings
On Time The time for which the input signal must be active and stable before a press is registered. <input type="text" value="0.02"/> s	Click A click is triggered if the debounced input is pressed for less than the Hold time.
On Hold Time The minimum length of time for which a press will be generated regardless of the input signal. <input type="text" value="0.02"/> s	Hold The minimum time for which the debounced input must remain pressed before a 'hold' condition is detected. <input type="text" value="0.75"/> s
Off Time The time for which the input signal must be inactive and stable before the signal is considered to be released. <input type="text" value="0.02"/> s	Long Hold The minimum time for which the debounced input must remain pressed before a 'long hold' condition is detected. <input type="text" value="1.50"/> s
Ignore Time The minimum length of time which must elapse between presses. <input type="text" value="0.05"/> s	

For most functions, the default values should work well.

Note: These settings are for all buttons on the **Inputs** tab. If you need to edit button settings, unselect the 'use default settings' option.

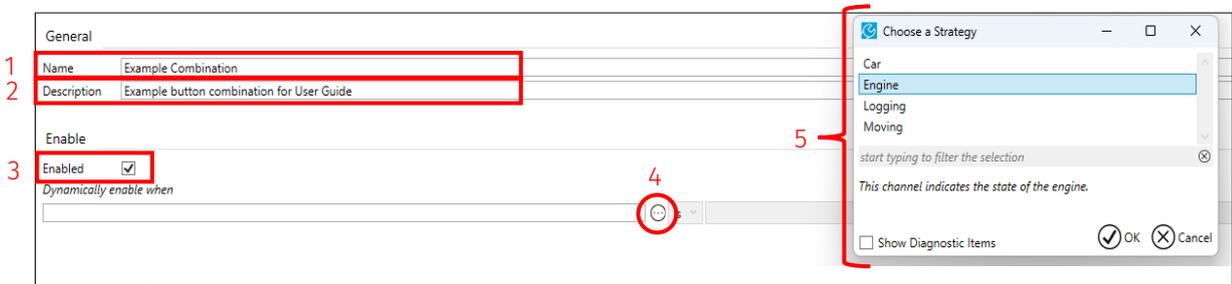
Configure button combinations

On the **Combinations** tab, you can configure channels that react from combinations of button inputs. For example, you can configure a channel to be triggered when two buttons are pressed simultaneously. This channel can be used to drive miscellaneous functionality within the setup.

To create a new button combination channel, click the **Combinations** tab (1), and then click the + button (2). Combinations can be deleted with the 'bin' icon (3).



You can configure the new button combination. Name the combination (1), and then provide an optional description (2). You can select if the combination is enabled in the setup (3). You can then use the **Choose a Strategy** option to select the strategy condition (4/5).



You can then define the button combination with options for the **Ordered** or **Unordered** button inputs:

- **Ordered:** the button needs to be pressed first, and then the second button to trigger the output channel.

Combination	
Mode	<input checked="" type="radio"/> Ordered <input type="radio"/> Unordered
Input	CCW Switch1
then input	CCW Switch2

- **Unordered** – Two buttons must be pressed to trigger the output channel but a time threshold ‘pressed within’ must be defined to trigger the channel.

Combination	
Mode	<input type="radio"/> Ordered <input checked="" type="radio"/> Unordered
Input	CCW Switch1
and input	CCW Switch2
pressed within	<input type="text" value="0.400"/> s

Configure a digital button

A button can also be configured from a digital input rather than from a CAN device. This is useful if you need an additional button in the setup which is separate to any CAN devices.

To configure a digital button, go to the **Sensors** node, and navigate to the available digital inputs. Click a digital input (1), click the import tool (2), select ‘Digital Push Button Sensor’ (3), and then click **Import** (4).

The screenshot illustrates the steps to configure a digital button. On the left, the 'Input Sensor Pairs' panel shows a list of digital inputs. 'Digital 01' is selected, indicated by a red box and the number 1. On the right, the 'Attach Sensor...' dialog is open, showing a search for 'Digital Push Button Sensor' (3) in the 'Digital Level Sensors' category. The 'Import' button is highlighted with a red circle and the number 4.



You can now configure the digital button. Enter a **Sensor Name** (1) and an optional comment about the digital button in the **Details** section (2). The actual button **Name** is configurable (3), as is the 'mode' (4), in the **Button** section. The 'mode' allows the user to configure if the button triggers on the rising or falling edge of the digital input channel.

Return to the **Buttons** node and acknowledge the creation of the digital button. The default button settings apply to this button, but default timings can be deselected. You can use the digital button in button combinations and in conjunction with CAN buttons.

Button structure and examples

The CSB or CCW buttons consist of an 8-bit output to determine what type of button press is commanded. These are configured as bit-fields by default with bit 0 to the left, and bit 7 to the right. You might need to acknowledge specific buttons press types and ignore others. The table below shows the press type and the output bit number and decimal value:

Press Type	Bit Number	Decimal Value
Raw	0	1
Debounced	1	2
Clicked	2	4
Held	3	8
Long Held	4	16
Click Latched	5	32
Latched Held	6	64



Latched Long Held	7	128
-------------------	---	-----

For example, to command a Maths channel to enable on a 'Latched Long Hold', the channel syntax is configured as shown below:

```
Equation _____  
Edit the equation that determines the value of this math channel.  
1 [Button 1] & 128 == 128
```

For this channel, when the button value is both equal to 128 (the '& 128' portion of the equation) **and** equal to 128 (the '== 128' portion), then the button output is acknowledged. Therefore, there is no chance that an intermediary button state triggers the channel output. The computed Maths Channel generates a value of 1 or 0.

