Logic channels overview

The **Logic Channels** node offers an alternative approach to configuring control channels, simplifying the creation of control channels that are complex to create in maths channels. The following covers the two features of the **Logic Channels** node – condition and counter logic.

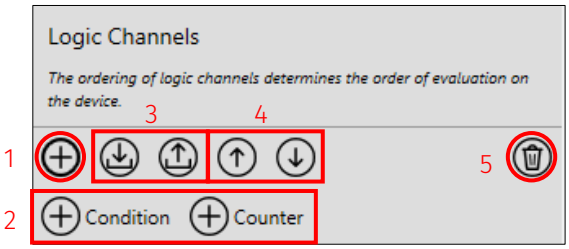
Note: Logic channels are limited to a fixed rate of 50Hz computation, but you can configure the computation rate of maths channels.

Add a logic channel

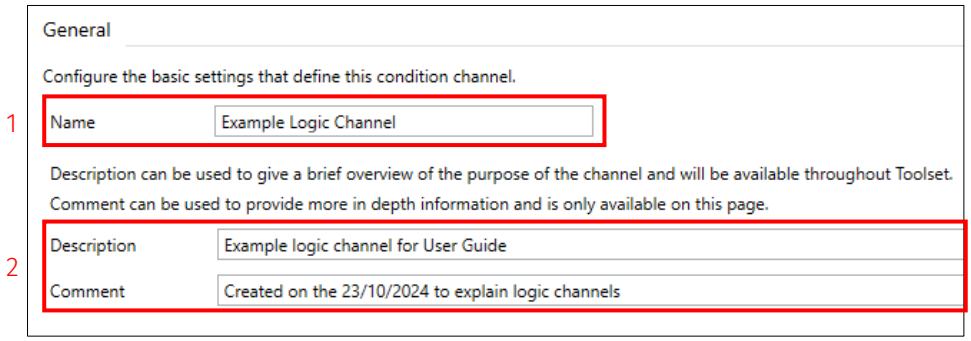
To add a new logic channel, hover over the **+** icon (1), and then click the required channel, either 'Condition' or 'Counter' (2). Use the 'import' and 'export' icons (3) to import and export logic channels between existing setups. If you add multiple logic channels, they can be reordered (4).

Note: - Logic channels are ordered in priority, so high priority and protected logic channels (see **Setup locking – channel protection**) must be at the top of the hierarchy.

Use the 'bin' tool to delete logic channels (5).



Enter a name for the channel (1) and an optional description and comment (2). Channel descriptions are used to provide a brief overview of the channel and are available to view when the logic channel is used elsewhere in Toolset. Channel comments are used to give more in-depth information about the channel and are only available to view on the **Logic Channels** node.

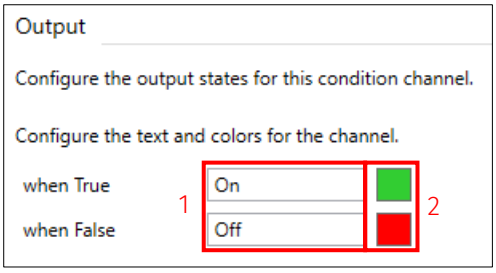


Conditional logic

In each conditional logic channel there are two main elements - logic groups and logic gates. Conditional logic passes input channels through 'logic groups' of 'logic gates' (also known as conditions). When the groups of conditions are true, the output is true.

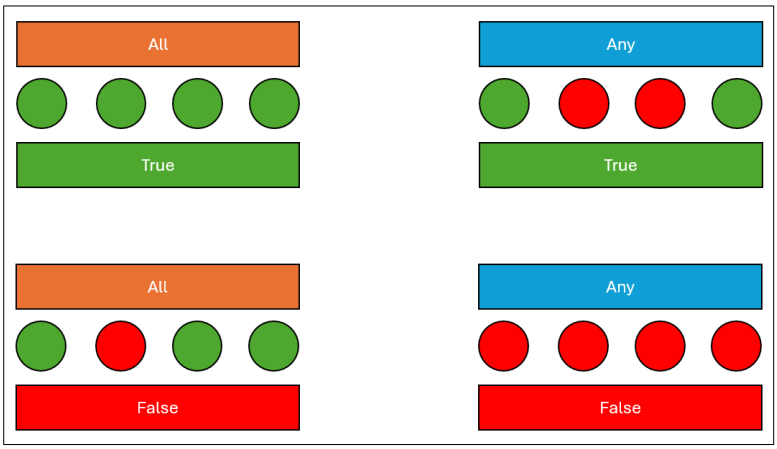
Generated channels

When you configure the conditional logic channel, an enumerated bit-field channel is generated. You can rename the true/false enumerated states for the bit-field channel (1) and set a colour for the true/false states (2).



Logic groups

Logic groups group conditions into ‘All’ or ‘Any’ to decide if all the conditions within a group must be met, or if only one condition within the group must be met, for the output to be true. In the logic tree, ‘All’ groups are highlighted in orange and ‘Any’ groups are highlighted in blue.



For example, the following logic channel is true when:

'Battery voltage >=13.5V' OR 'Switch 1 is pressed And Switch 2 is not pressed'.

Logic conditions

These are the channel conditions that must be true for the output to be true.

There are many types of condition, summarised in the table below. In the logic 'tree', logic conditions are highlighted in green.

Condition	Inputs	Description
Alarm Triggered	1x Alarm	A logical output based on an alarm state (see Alarms)
Bitfield Comparison	1x Bit-field Channel	A logical output based on a bit-field channel condition
Channel Comparison	1x Channel	A logical output based on a channel condition
Strategy Comparison	1x System State	A logical output based on a system state (see System States)
Logical AND	2x Channels or System States	Comparing 2 channels (If Chan1 AND Chan2)
Logical OR	2x Channels or System States	Comparing 2 channels (If Chan1 OR Chan2)
Logical XOR	2x Channels or System States	Comparing 2 channels (If Chan1 XOR Chan2)
Bitwise AND	1x Channels or System States	Compare a channel to a present hexadecimal value as either a binary or hexadecimal value
Flash	1x Channels or System States	While a channel is active the output flashes (on/off) at a preset rate
Hysteresis	1x Channels/ System States	Active when a set value is exceeded and deactivated when another different limit is reached

Pulse	1x Channels or System States	On a channel edge the output will pulse for a pre-set time
Set/Reset	2x Channels or System States	A channels edge drives this high and another drives this low
Toggle	1x Channels or System States and 2 Override Channel inputs	A channel changes this state on each pulse. Channels can be reset to override this value

Conditional logic channel example

The following example of a conditional logic channel is used to control a reserve fuel pump.
It is activated if ANY of these conditions are met:

- The Ignition is pressed AND runs for 5 seconds.

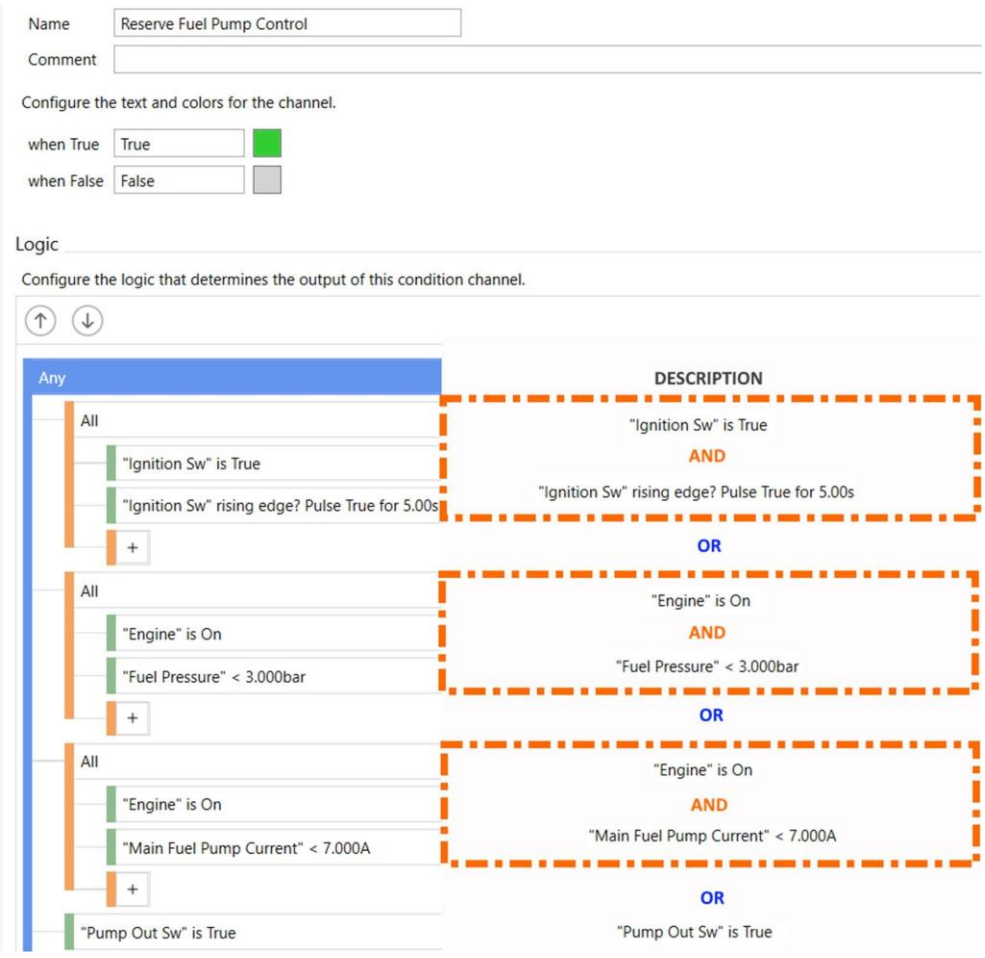
- The Engine is running AND Fuel Pressure is less than 3 bar.

OR

- The Engine is running AND the Main Fuel Pump Current is less than 7 amps.

OR

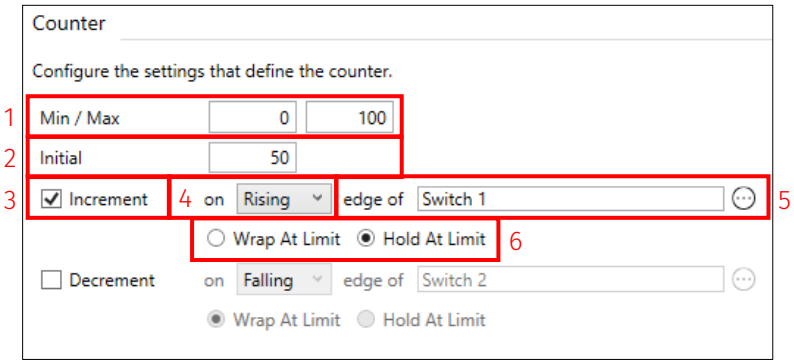
- The Pump Out switch is pressed.



Counter logic

Counter logic channels offer an alternative approach for you to create a numerical counter and avoid register-based maths channels, which can be complex for new users. Counter logic channels allow you to define a counter to be incremented or decremented based on either the rising or falling edge of a channel, or system state inputs.

To configure a counter, you need to set minimum and maximum limits (0 to 65535) for the counter (1). You also need to see the initial starting value of the counter (2). You can choose to increment and/or decrement the counter by selecting either or both increment and decrement check boxes (3). Configure the counter to increment or decrement on the rising or falling edge of the channel or system state from the dropdown menu (4). You can then select the channel or system state from the 'channel browse' menu (5). Finally, you can configure the counter **Wrap At Limit** (to roll over the counter to minimum or maximum value when the maximum or minimum limit is reached), or **Hold At Limit** (hold the counter at the minimum or maximum limit when the limit is reached) (6).



You can configure force counter values if defined conditions are met. Select the two from the **Set** check box (1). You can set the value to override the counter (2), and then define the channel condition when the override happens (3).