

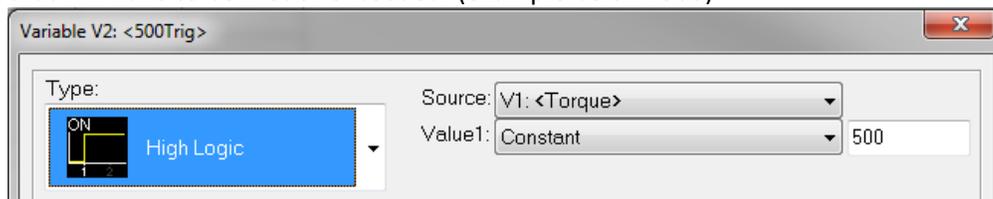


Quick Start Guide: Creating a “One-Shot Trigger” using the ValueChanged and HigherEqual Functions within test.commander

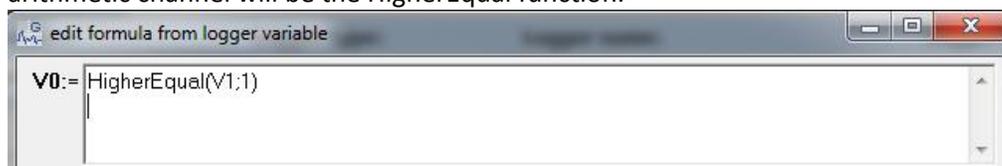
Purpose: This document describes how to create a “one shot trigger” to start logging a data set based on a certain level being met or exceeded. This type of trigger uses a combination of 2 x built-in arithmetic functions: ValueChanged and HigherEqual.

Procedure:

1. Setup the input channel to monitor the process.
2. Create an alarm channel to monitor the input channel. Select the value of the input channel that will have to be met or exceeded. (example below: 500)



3. When the value hits 500 or more, the alarm channel will output “1”.
4. A data logger can be triggered based on this alarm, but if the value dips below the threshold, the alarm will also switch from “1” to “0”. Certain processes need to simply start the logging for a pre-set amount of time while other processes log as long as the condition is true. Using the ValueChanged and HigherEqual combination, it is possible to create this one time trigger for a pre-set amount of time.
5. Create a new Arithmetic channel under the Virtual Variables section of the controller or if using a Q.station, an Arithmetic channel inside the logger configuration can be created. The first arithmetic channel will be the HigherEqual function.



6. Using the example above, V1 is the alarm channel we are monitoring. If this alarm channel’s value is equal or greater than “1”, the value of the HigherEqual variable will change from “0” to “1”. More information about the HigherEqual function:

Compares 2 x arguments and if Arg1 is higher or equal to Arg2, the result is 1, otherwise it is 0.

Format: HigherEqual(Arg1;Arg2)

The arguments can be a variable or a constant value.

Example:

HigherEqual(35;35) = Result: 1

HigherEqual(17;35) = Result: 0



7. Create a new Arithmetic channel under the Virtual Variables section of the controller or if using a Q.station, an Arithmetic channel inside the logger configuration can be created. The second arithmetic channel will be the ValueChanged function.



8. There are several ways to use the ValueChanged function, but the most common is used above. Here V1 is the HigherEqual variable being monitored. If the value of this channel rises from "0" to "1" (aka a rising edge detection), the result of this function will equal "1" for 1 x cycle.

Format: ValueChanged(Variable;Type;HandleValue)

Variable – The value to be monitored.

Type –

- 0: The parameter of the accuracy can be defined. Example: 0.1, if the value of the Variable changes by 0.1, this will indicate an event and the value of the function will result in "1" for 1 cycle.
- 1: Step since last cycle is greater than HandleValue. Example: 0.5, rising edges can be detected. If the Variable changes from 0 to 1, the value of the function will result in "1" for 1 cycle.
- 2: Step since last cycle is lower than HandleValue. Example: 0.5, falling edges can be detected. If the Variable changes from 1 to 0, the value of the function will result in "1" for 1 cycle.

9. The logger can now be triggered base on the ValueChanged function. For example, if ValueChanged > 0.5, the logger will be started. It will log data for the pre-defined logging length (including any pre-trigger specifications).

10. The example below shows rising edge detection and the ValueChanged (green line) variable changes to "1" for 1 cycle even though the alarm condition is still true.



Contact us today if you have any further questions!