

Lorby “Axis And Ohs” StreamDeck Plugin Documentation

Version 4.42 - 24.09.2021

© 2021 Lorby-SI <http://www.lorby-si.com>

This plugin for the ElgatoStream deck uses the AxisAndOhs WebAPI.

Make sure to activate it first:

- AAO must be started “As Administrator”
- Then you can activate “Enable Web API on port 9080” in the “Tools” menu

You may have to open/allow access to the port in your local firewall. The port number can be changed with “Tools->Port settings for Web” in case there is a conflict with another app on your computer. A restart of the app is necessary when you change the port number.

If you have another app on your computer that is already using 9080, the port must be changed in the AAO Tools menu **and** in the settings of the plugin:

C:\Users\...\AppData\Roaming\Elgato\StreamDeck\Plugins\com.lorbysi.aao.sdPlugin\settings.js

The file can be edited with a simple text editor, and it contains only one single line where you can just alter the port number:

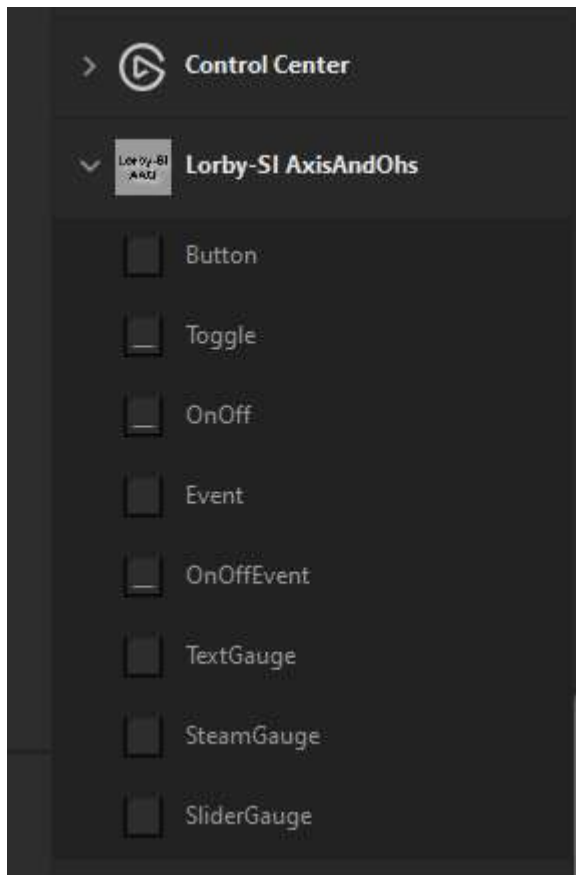
```
const AAO_URL = "http://localhost:9080/webapi";  
const REFRESH_MS = 100;
```

If you are experiencing performance problems with the Elgato app or the StreamDeck, you can slow down the refresh rate of the plugin with the second parameter “REFRESH_MS”.

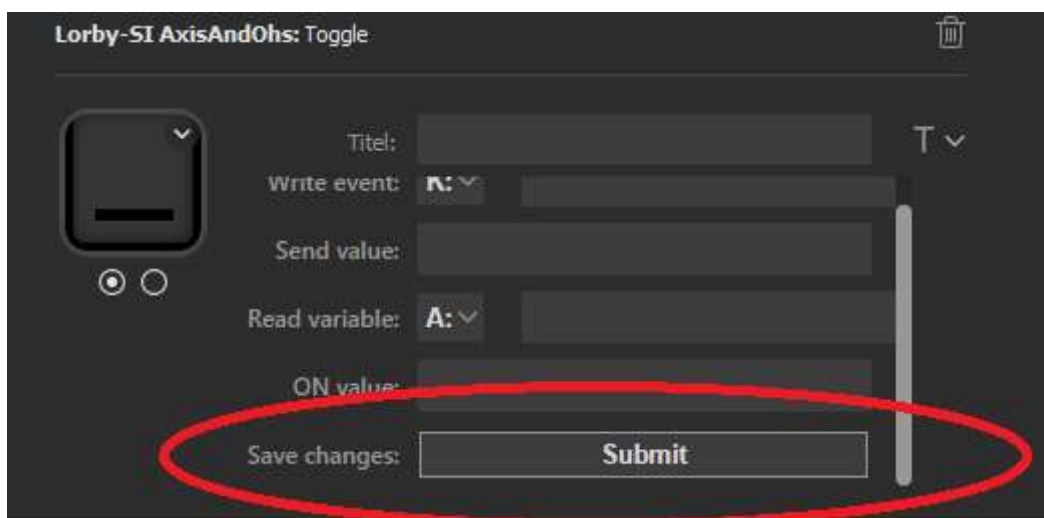
- A value of 1000 equals one refresh every full second.
- Don't go below 100 (refresh happens 10 times a second)

The Elgato software must be shut down and restarted after these changes.

The Plugin provides several different types of buttons for the StreamDeck.



Every button type that you place on the stream deck has properties assigned to it. When you change those properties, make sure to press the “SUBMIT” button at the bottom of the property inspector area in the Elgato software, otherwise your changes will not be applied to the button

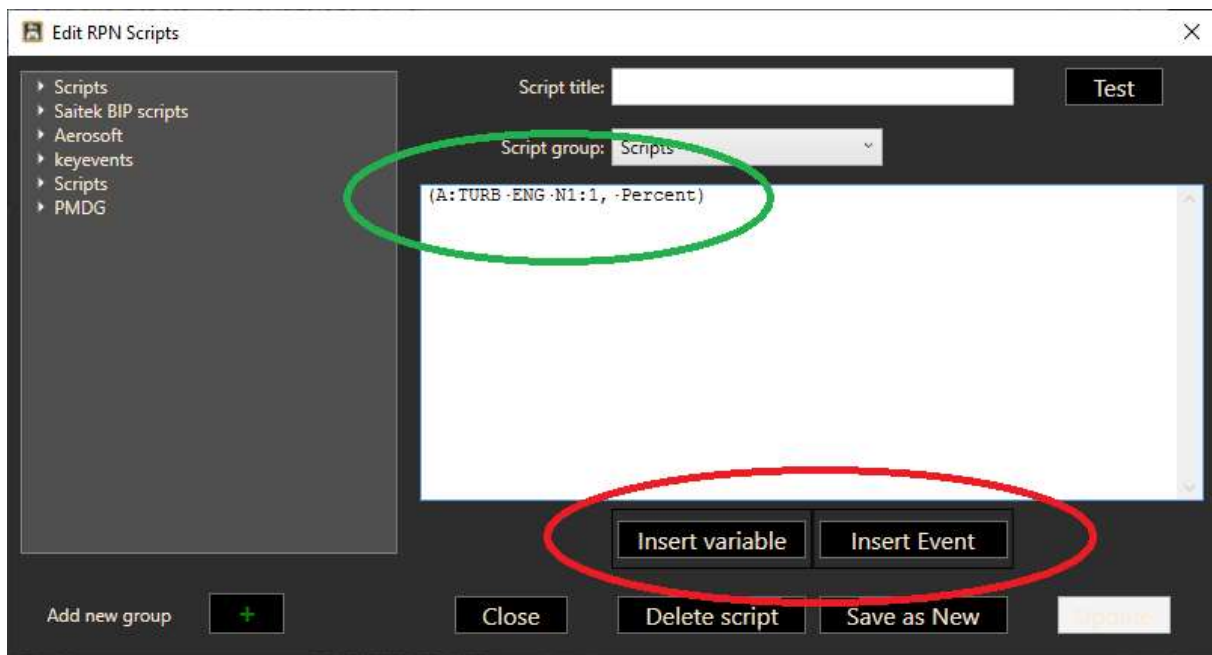


Simulator Events and Variables

This plugin will use the events and variables that are defined for the compatible simulators.

You will need either the SDK documentation for the simulator or make use of the RPN script editor in AxisAndOhs.

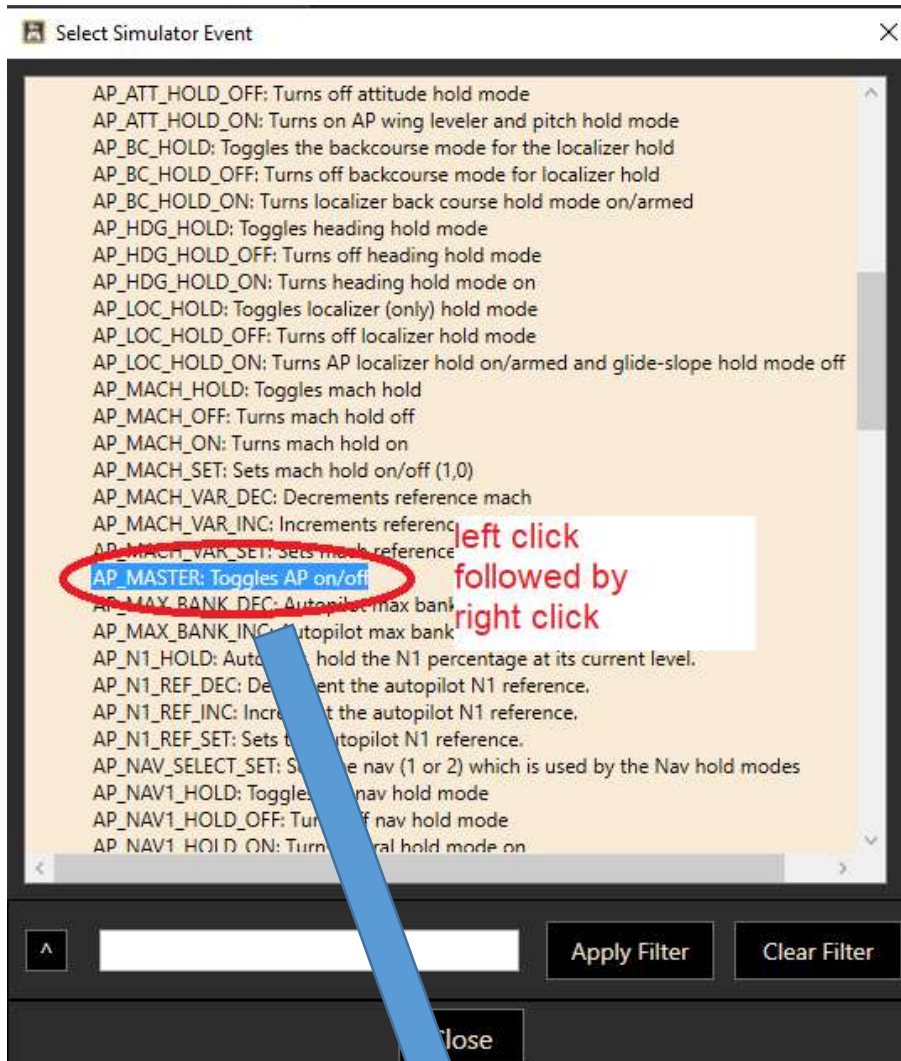
In AxisAndOhs you go to “Scripting -> Manage RPN scripts”. Use “Insert variable” / “Insert event” to find out the exact syntax of the required asset.



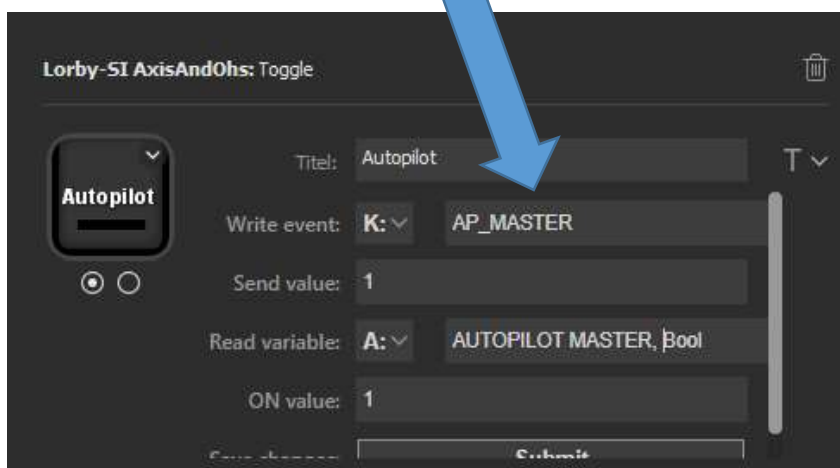
You can also request values that have been calculated by an AAO RPN script, using

- „L: scriptgroup-scriptname“ for numerical values, calculated by a script like this one:
„(A:INDICATED·ALTITUDE,·Feet)·10000·+“
- „L scriptgroup-scriptname, String“ for string values that are produced by a script like this:
„%(A:INDICATED·ALTITUDE,·Feet)·10000·+%!5d!

When the variable/event lists are open, you can copy the name of the variable or event to the Windows scratchpad by left clicking then right clicking the item in the treelist. Then you can insert it into the textbox in the Elgato software by pressing Ctrl&V

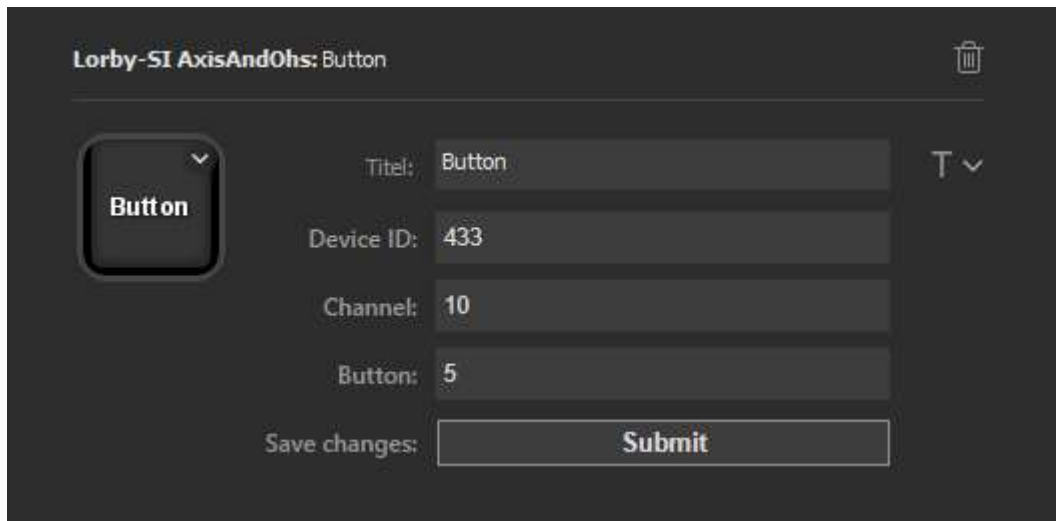


CTRL & V




Type “Button”

This simulates a button press on a Joystick.



The screenshot shows a configuration window titled "Lorby-SI AxisAndOhs: Button". On the left is a button icon labeled "Button". To the right are four input fields: "Title:" with the value "Button", "Device ID:" with "433", "Channel:" with "10", and "Button:" with "5". Below these fields is a "Save changes:" label and a "Submit" button. A trash icon is in the top right corner, and a "T" icon with a dropdown arrow is on the right side of the input fields.

In AxisAndOhs, pressing the button will be registered on the “Add/Change” button dialogs like this:

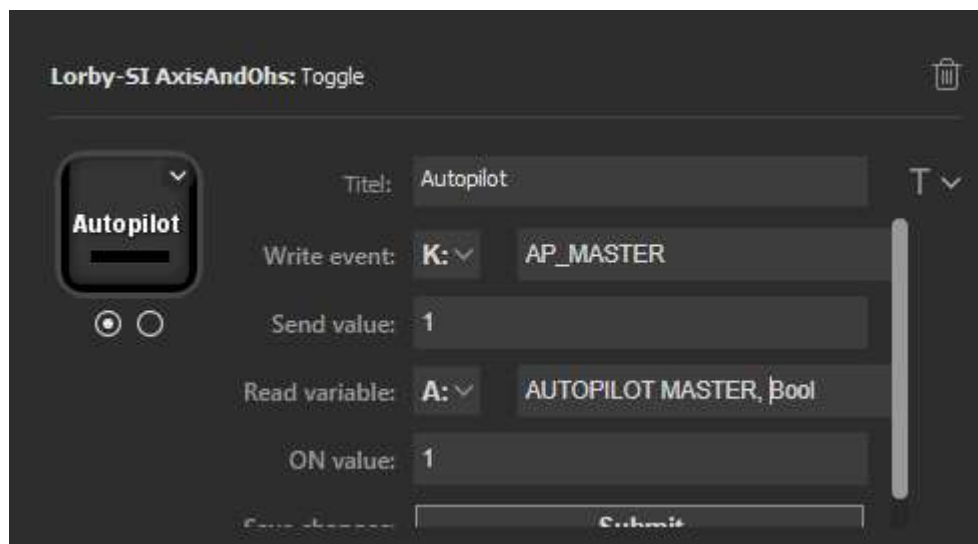


The screenshot shows a table with three rows. The first row has a green circle icon, the text "Virtual key:", and a dropdown menu set to "None". The second row has a green circle icon, the text "Assigned button/key:", and a text box containing "WebAPI: 433 CHN: 10 EVT: 5". Below this text box is a "Device:" label and a dropdown menu set to "All". The third row has a green circle icon and the text "None". Each row has a red "X" icon in a small box on the right.

Virtual key:	None	X
Assigned button/key:	WebAPI: 433 CHN: 10 EVT: 5 Device: All	X
	None	X

Type “Toggle”

If the desired simulator action is bound to a single toggle event, use this button type to switch between the two states. A simulator variable can be used to receive feedback about the success of the toggle. When the ON value is received, the plugin will load the “on” picture for the button, otherwise the “off” picture is set.



Type “On/OFF”

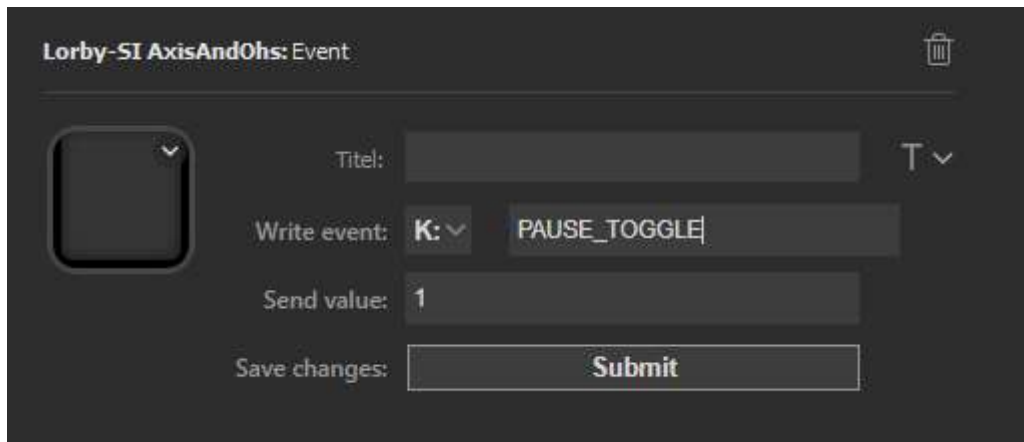
In this case there is no toggle event in the simulator, but separate ON and OFF events instead:

The image shows a configuration window for a device named "Lorby-SI AxisAndOhs: OnOff". On the left is a visual representation of the device with a green bar and two circular indicators. The main area contains configuration fields:

- Titel:** A text input field.
- Write event:** A dropdown menu set to "K:" followed by a text field containing "AP_HDG_HOLD_ON".
- ON value:** A text input field containing "1".
- Write event:** A dropdown menu set to "K:" followed by a text field containing "AP_HDG_HOLD_OFF".
- OFF value:** A text input field containing "1".
- Read variable:** A dropdown menu set to "A:" followed by a text field containing "AUTOPILOT HEADING LOCK, Bool".
- ON value:** A text input field containing "1".
- Save changes:** A button labeled "Submit".

Type “Event”

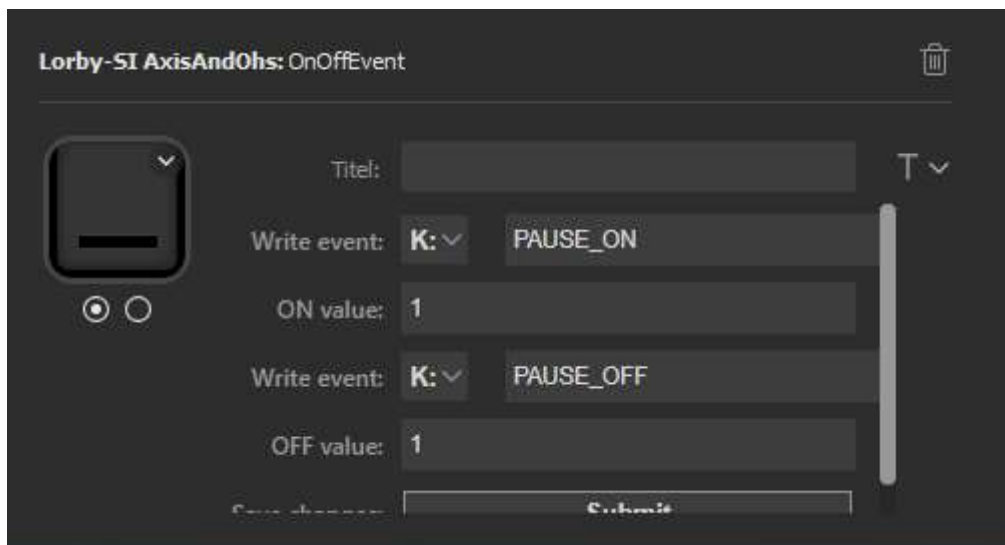
Send a simulator event directly without querying a variable for a result



The screenshot shows a configuration window titled "Lorby-SI AxisAndOhs: Event". On the left is a square icon with a downward arrow. To the right of the icon are several input fields: "Titel:" followed by a text box, "Write event:" followed by a dropdown menu showing "K:" and a text box containing "PAUSE_TOGGLE", "Send value:" followed by a text box containing "1", and "Save changes:" followed by a "Submit" button. A trash icon is in the top right corner.

Type “DualEvent”

Send separate ON/OFF simulator events directly without querying a variable for a result



The screenshot shows a configuration window titled "Lorby-SI AxisAndOhs: OnOffEvent". On the left is a square icon with a horizontal line and two radio buttons below it. To the right of the icon are several input fields: "Titel:" followed by a text box, "Write event:" followed by a dropdown menu showing "K:" and a text box containing "PAUSE_ON", "ON value:" followed by a text box containing "1", "Write event:" followed by a dropdown menu showing "K:" and a text box containing "PAUSE_OFF", and "OFF value:" followed by a text box containing "1". A "Submit" button is at the bottom. A trash icon is in the top right corner.

Note: all of the following “Gauge” type actions can also be used to trigger events. At the end of the parameter list you will find a “Button event” box for K:, L: and H: events/Variables with the “Send value” right below. When you press the gauge on the StreamDeck, it will send this event/ change this variable.

Type “TextGauge”

This button type will display the values of one or two simulator variables as text on the button

Lorby-SI AxisAndOhs: TextGauge

Titel: MSL AGL

Variable 1: A: Indicated Altitude, feet

Multiplier: 1

Decimals: 0

Shift Y: 45

Variable 2: A: PLANE ALT ABOVE GROUND, Feet

Multiplier: 1

Decimals: 0

Shift Y: 95

Font: Arial Black

Font: Arial Black

Font Size: 50

Text Color: ■

Back Color:

Background Image:

Button event: K: PAUSE

Send value: 1

Save changes:

- **Multiplier:** multiply the variable value with this factor before displaying it
- **Decimals:** number of decimal fraction digits to display
- **Leading 0s:** pad the number to the left with '0' until this length has been reached. Example: a value of 4 makes the number 12 look like 0012.
- **Shift Y:** vertical position in pixels where the text is to be displayed
- **Background Image:** Use 144x144 pixel image files in PNG format.

Do not use the standard Elgato picture upload (the little arrow on the button icon), because this will override the text display.

Type “SteamGauge”

A “Steam Gauge” button displays a graphical gauge with a rotating element. An additional text can be displayed for the variable value.

The gauge consists of three layers:

- **Background (optional)::** the static background picture
- **Turning:** the rotating part, painted on the background
- **Mask (optional)::** another static picture that is painted on top of the other two

You can use any PNG picture with 144x144 pixel maximum size and transparency.

- **Read variable:** simulator variable to read the rotation angle from. The value must be in **Radians**.
- **Multiplier:** multiply the variable value with this factor before applying it .
- **Turn offset deg:** value (in Degrees!) to apply as the initial turn (=when the variable value is 0)
- **Shift X, Shift Y:** location of the center point for the rotation in pixels.



Titel:



Read variable:

A: ▾

PLANE HEADING DEGREES MAG

Multiplier:

1

Turn offset deg:

0

Shift X:

72

Shift Y:

72

Text Variable:

A: ▾

PLANE HEADING DEGREES MAG

Multiplier:

1

Decimals:

0



Shift Y:

85



Font:

Tahoma



Font Size:

50

Text Color:



Back Color:



Upload Background

Clear



Upload Background

Clear

Upload Turning

Clear

Upload Mask

Clear

Save changes:

Submit



Upload Mask

Clear

Button event:

K: ▾

Send value:

Save changes:

Submit

Type “SliderGauge”

A “Slider Gauge” button uses the same layered method as the “Steam Gauge”, but instead of rotating the middle picture, it is shifted along the X and Y axis, or both.

The gauge also consists of three layers:

- **Background (optional)::** the static background picture
- **Sliding:** the moving part, painted on the background
- **Mask (optional):** another static picture that is painted on top of the other two

You can use any PNG picture with 144x144 pixel maximum size and transparency.

- **Read variable:** simulator variable to read the value from.
- **X Multiplier, Y Multiplier:** multiply the variable value with this factor before applying it to the associated movement axis.
- **X Offset, Y Offset:** value in pixel to apply as the initial offset (=when the variable value is 0)
- **Shift X, Shift Y:** location of the center point for the rotation in pixels.



Titel: FLAPS



Read variable: A: FLAPS HANDLE PERCENT, Percent

X Multiplier: 1,44

Y Multiplier: 0

X Offset: 80

Y Offset: 0

Y Offset: 0

Text Variable: A: FLAPS HANDLE PERCENT, Percent

Multiplier: 1

Decimals: 0

Shift Y: 85

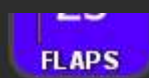
Font: Tahoma

Font Size: 50

Text Color:



Back Color:



: Upload Background

Clear

: Upload Sliding

Clear

: Upload Mask

Clear

Save changes: Submit

: Upload Mask

Clear

Button event: K:

Send value:

Save changes: Submit