

Lorby „Axis And Ohs“ für den Microsoft Flugsimulator Dokumentation

<https://www.axisandohs.com/>

Englische Version 4.57 – 19.05.2026

© 2021 LWR Inc.

Übersetzung DVRGL, Version 4.56 – Juni 2026

(unter Verwendung von DeepL)

Inhaltsverzeichnis

1 Übersicht.....	6
2 Installation.....	7
2.1 Bereitstellung.....	7
2.2 Installation.....	7
3 Betrieb.....	9
3.1 Start der Anwendung.....	9
3.2 Hauptfenster.....	10
3.3 Hauptmenü.....	11
3.4 Haupt- und Clonekonfiguration.....	13
3.5 Steuerungsebenen.....	15
3.6 Sperren der GUI-Bedienelemente zur Vermeidung versehentlicher Änderungen.....	17
3.7 Verwaltung der Joystick-Achsenbelegung.....	18
3.7.1 Eine neue Achse zuweisen.....	18
3.7.2 Empfindlichkeit der Joystick-Achsen.....	22
3.7.3 Achsen Kalibrieren.....	23
3.7.4 Axis-Erweiterter Modus.....	24
3.7.5 Eine Achsenzuordnung ändern.....	25
3.7.6 Einer Simulatorvariablen eine Achse zuweisen.....	26
3.7.7 Einmalige und virtuelle Ereignisse.....	26
3.7.8 Voreingestellte Kombination.....	28
3.7.9 Achse kopieren oder entfernen.....	30
3.8 Verwaltung von Tasten-, MIDI- und Tastaturzuweisungen.....	31
3.8.1 Eine neue Taste zuweisen.....	31
3.8.2 Auswahl vs. manuelle Zuweisung.....	32
3.8.3 Der K:2:index.....	32
3.8.4 Parameter für die Buttonbetätigung.....	33
3.8.5 Werte an Ereignisse senden.....	34
3.8.6 Virtuelle Ereignisse.....	34
3.8.7 Voreingestellte Kombination.....	35
3.8.8 Langer Klick (Long click).....	37

3.8.9 Die Steuerung der Tastenbelegung in der Hauptliste.....	37
3.8.10 Tastenbelegung ändern.....	38
3.8.11 Zuweisungen für die Schaltflächen Kopieren oder Entfernen.....	39
3.9 Ereignisauswahlfelder verwenden.....	40
3.10 Verwendung von Templates.....	41
3.11 Ansichtsmodi des Bedienfelds.....	46
4 Spracherkennung.....	47
5 Skripting.....	58
6 Verwendung anderer Skriptsprachen als RPN.....	113
7 MSFS-Eingabeereignisse.....	122
8 Das AxisAndOhs In-Game-Panel.....	123
9 Maus als Steuerknüppel.....	128
10 Erweiterte Energieverwaltung (Win 8.1 und höher).....	130
11 Saitek-Panels.....	131
12 Desktop-FIPs.....	137
13 Web-FIPs.....	140
14 App- und Web-Fenster.....	143
15 Hardware-Änderung.....	145
16 Geräte-Blacklist.....	148
17 Konfigurationsdateien, Datenbank und Protokolldateien.....	149
18 PMDG-Flugzeuge.....	151
19 MIDI-Ausgang.....	155
20 Mehrere MIDI-Geräte – Kanalpriorität.....	157
21 Web-API.....	158
21.1 URL.....	158
21.2 Ändern der Ports.....	159
21.3 Senden eines simulierten Schaltflächenereignisses.....	160
21.4 Senden eines simulierten Achsereignisses.....	161
21.5 JSON-Schnittstelle.....	162
22 Verwendung der API als Webserver.....	168
23 Ereignis- und Variablenlisten importieren.....	173
24 Text-to-Speech: WinRT vs. SAPI.....	174

25 Erweiterte Text-to-Speech-Funktionen: Azure, Polly, ChatGPT.....	176
26 RPN-Skriptdateien.....	180
27 Interaktive Text-to-Speech-Checkliste.....	182
28 Der Dialog „Checklist Creator“	187
29 Verschleißsimulation.....	191
30 Zuordnung von LVar zu BVar (veraltet, bitte nicht verwenden).....	198
31 vJoy-Schnittstelle.....	205
32 ViGEm-Schnittstelle.....	206
33 Virtuelle Maus.....	207
34 CAN-Schnittstelle.....	208
35 Soundeffekte für Audio- und Sprachausgabe.....	211
36 ChatGPT-Schnittstelle.....	214
37 Befehlszeilenparameter.....	217
38 Haftungsausschluss.....	218

**Absichtlich frei gelassen um Seitengleichheit
mit dem Original zu erhalten.**

1 Übersicht

Die Lorby-SI-App „AxisAndOhs“ wurde entwickelt, um Ihre Joysticks und andere Controller individuell und automatisch für jedes Flugzeug zu verwalten, das Sie in Ihrem Simulator fliegen. AxisAndOhs speichert jede Joystick-Belegung für jede Flugzeuglackierung. Wenn Sie bereits eine Belegung für ein anderes Flugzeug dieses Typs in der Datenbank haben, übernimmt AxisAndOhs diese Belegung und wendet sie an.

Alle Joystick-Bewegungen werden dann über die App geleitet und steuern Ihr Flugzeug im Simulator direkt. Es sind keine weiteren Steuerungszuweisungen erforderlich, weder im Simulator noch über ein externes Modul.

2 Installation

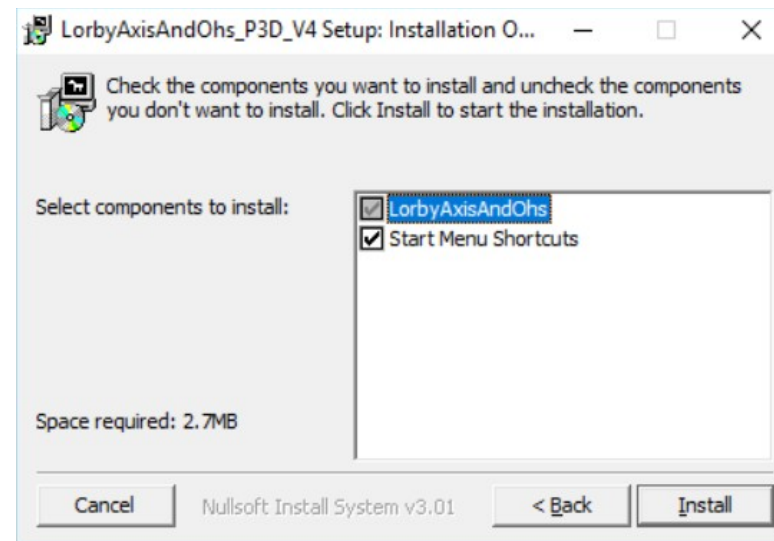
2.1 Bereitstellung

Lorby AxisAndOhs wird als selbstextrahierendes Installationspaket bereitgestellt.

2.2 Installation

- Für diese Anwendung müssen die .NET 4.8-Laufzeitumgebung und die VC++ Redistributable 2019 auf Ihrem Computer installiert sein.
<https://dotnet.microsoft.com/download/dotnet-framework/net48>
<https://support.microsoft.com/en-us/help/2977003/the-latest-supported-visual-c-downloads>
- Bitte verwenden Sie das Installationsprogramm „LorbyAxisAndOhs_Install_MSFS.exe“

Starten des Installers:



Auf der ersten Seite können Sie optionale Installationsziele auswählen:

- „Startmenü-Verknüpfungen“: Lorby AxisAndOhs wird Ihrem Startmenü hinzugefügt (empfohlen)
- Wenn Sie „Installieren“ auswählen, wird die Installation gestartet

3 Betrieb

3.1 Start der Anwendung

- Starten Sie die Lorby AxisAndOhs app
Klicken Sie auf die grüne LED in der oberen Menüleiste
oder öffnen Sie das Menü „Verbindung“ und wählen Sie „Verbinden“
oder stellen Sie die App im selben Menü auf „Automatisch verbinden“ ein
- Starten Sie den Microsoft Flight Simulator
- Sobald Ihr Flugzeug erkannt wurde und die Simulation läuft, können Sie mit dem Hinzufügen von Achsen-Steuerelementen und Schaltflächen beginnen. Die LAAO-App speichert diese Einstellungen für jedes Flugzeug.
Falls die App das Flugzeug nicht automatisch erkennt, wenn Sie im Cockpit sitzen, verwenden Sie bitte die Option „Verbindung zur laufenden Simulation erzwingen“ im Menü „Verbindung“.
- Um Konflikte mit den Reglereingängen zu vermeiden, sollten Sie widersprüchliche Zuweisungen auf der Seite „Reglerkonfiguration“ der Simulatoren deaktivieren

3.2 Hauptfenster



- Obere Beschriftung: Hier wird der Name der aktuellen Konfiguration angezeigt. Bei Verbindung mit dem Simulator erkennt die App das Flugzeug und lädt die Konfiguration automatisch. Sie können auf diese Beschriftung klicken, um eine andere Konfiguration oder Vorlage zu laden.
- Zugewiesene Achsen: Liste aller Joystick-Achsen, die Sie diesem Flugzeug zugewiesen haben
+/-: zum **Hinzufügen** oder **Entfernen** einer Achsenzuweisung.
- Zugewiesene Tasten: Alle Tastenbelegungen (Joystick, MIDI und Tastatur), die Sie diesem Flugzeug hinzugefügt haben
+/-: zum **Hinzufügen** oder **Entfernen** einer Tastenbelegung.
- Belegungen für nicht angeschlossene Controller werden rot hervorgehoben



Wenn Sie das Mausrad über den Zwischenraum zwischen den beiden Listen drehen, ändert sich die Größe des Fensters. Ein Rechtsklick auf den Zwischenraum setzt die Größe wieder auf die Standardeinstellung zurück.

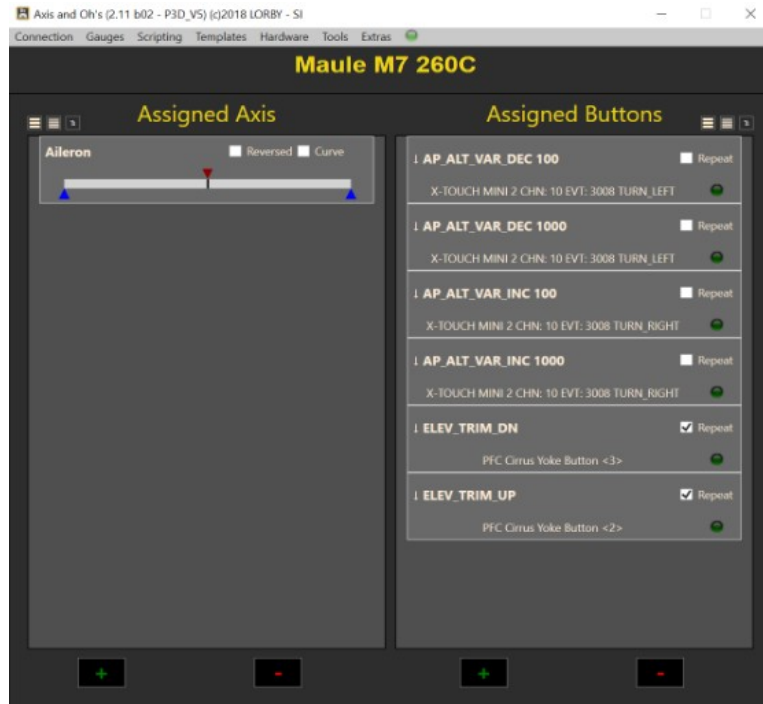
3.3 Hauptmenü

- **Anwendung (Application)**
Allgemeine Einstellungen
Konfigurieren Sie die App so, dass sie beim Start automatisch eine Verbindung herstellt und/oder minimiert wird.
Konfigurieren Sie die App so, dass sie in die Taskleiste statt in den Infobereich minimiert wird.
Starten Sie den Simulator automatisch vor dem Herstellen der Verbindung.
Starten Sie die App im Offline-Modus, mit oder ohne vorgewählte Konfiguration.
- **Anzeigen (Gauges)**
Aktivieren Sie Desktop- und Web-Fluginstrumente
(für WebFIPs muss die App als „Administrator“ gestartet werden)
- **Skripting (Scripting)**
RPN-Skripte verwalten/bearbeiten
- **Vorlagen (Templates)**
Eine vorhandene Konfiguration Ihrem aktuellen Flugzeug zuweisen
Vorlagen für Flugzeugkonfigurationen erstellen und verwalten
- **Hardware**
Saitek-Geräte anschließen und verwalten, MIDI-Geräte anschließen, Mausempfindlichkeit, Geräte-Blacklist und Hardwareänderungen verwalten

- **Werkzeuge (Tools)**
Auftragsdatenbank speichern, EPM verwalten, Web-Ports ändern, Schaltfläche suchen
- **Extras**
Spezielle Funktionen, die normalerweise nicht im Simulator vorhanden sind
- **SimConnect-Modus – Grüne LED**
durch Klicken auf die LED eine Verbindung zum Simulator herstellen bzw. trennen
- **Offline-Modus – Rote LED**
Im Offline-Modus stellt AAO keine Verbindung zum Simulator her, aber die Eingabegeräte funktionieren alle. Sie können eine Konfiguration oder Vorlage manuell laden oder eine neue erstellen und diese bedienen. Beachten Sie, dass die Simulator-Variablen und Ereignisse in diesem Modus nicht verfügbar sind.
Im Offline-Modus kann die App als „Brücke“ zwischen Ihren Eingabegeräten und jeder anderen Software dienen, die Joystick- oder Gamepad-Eingaben akzeptiert, wobei die vJoy- und ViGEM-Schnittstellen verwendet werden.
Sie könnten beispielsweise Ihr Flugsimulator-Steuerhorn als Lenkrad in SnowRunner verwenden, obwohl das Spiel nur Eingaben von Xbox-Controllern akzeptiert. Weisen Sie einfach die Achse Ihres Steuerhorns der entsprechenden ViGEM-Achse zu.
 - Um automatisch im „Offline-Modus“ zu starten, aktivieren Sie „Anwendung->Automatischer Offline-Modus“
 - Um eine bestimmte Konfiguration zum automatischen Laden auszuwählen, laden Sie diese zunächst in AAO und aktivieren Sie dann die Option „Anwendung->Konfiguration laden:“.

3.4 Haupt- und Clonekonfiguration

Die erste Konfiguration, die Sie für ein Flugzeug erstellen, gilt als „Master“. Wenn Sie dann dasselbe Flugzeug, jedoch mit einer anderen Lackierung laden, geht AAO davon aus, dass Sie dieselben Steuerelemente wie beim Master verwenden möchten.



MASTER



CLONE

Wenn eine Konfiguration wie oben beschrieben „geklont“ wird, sind alle Axis- und Buttons-Assets gesperrt – du kannst sie nur in der „Master“-Konfiguration ändern!

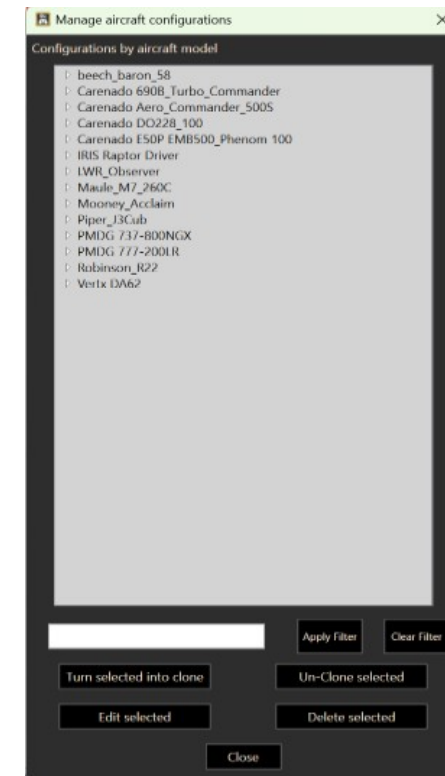
Falls die automatische Erkennung eines Klons nicht funktioniert, kannst du das Flugzeug zwingen, eine bestimmte Master-Konfiguration zu verwenden, indem du „Vorlagen → Master-Konfiguration für dieses Flugzeug auswählen“ wählst.

Wenn du eine separate Konfiguration für das Flugzeug speichern möchtest, deaktiviere das Kontrollkästchen in der roten Leiste. AAO lädt die Konfiguration dann als separate Einheit neu.

Um eine Konfiguration zurückzusetzen, verwende „Vorlagen → Aktuelle Konfiguration löschen“

Um die Master-/Klon-Beziehungen zu verwalten, verwende „Vorlagen → Konfigurationen bearbeiten“

- Jedes Flugzeug muss mindestens eine „Master“- Konfiguration haben
- Die anderen können mit der Schaltfläche unterhalb der Liste in „Klone“ umgewandelt werden.



3.5 Steuerungsebenen

Schaltflächen und Achsen können in 4 Ebenen angeordnet werden. Eine Schaltfläche oder eine Achse wird nur dann „ausgelöst“, wenn Sie die entsprechende Ebene in AAO auswählen. Achsen und Schaltflächen sind in separate Ebenenfolgen unterteilt. Schaltflächen und Achsen, die nur der Standardebene „0“ zugewiesen sind, werden immer ausgelöst, unabhängig davon, welche Steuerungsebene gerade eingestellt ist.



(vorne im farbigen Bereich anklicken um den Layer zu ändern)

Ebene 0 – Standard, keine Farbe: Die Taste oder Achse wird immer ausgelöst (auch wenn eine andere Ebene für AAO ausgewählt wurde), es sei denn, sie ist auch auf anderen Ebenen vorhanden.

Ebene 1 – gelb, 2 – rot, 3 – grün usw.: Schaltfläche oder Achse wird nur ausgelöst, wenn diese Steuerungsebene für die AAO-App ausgewählt wurde. Die ersten drei Ebenen sind fest codiert. Sie können zusätzliche Ebenen über den Dialog „Vorlagen->Steuerungsebenen verwalten“ erstellen.

Um eine Ebene auszuwählen, können Sie

- entweder auf die Beschriftungen unterhalb der Achsen- und Schaltflächenfelder klicken
- oder auf den Bereich ganz links im Schaltflächen- oder Achsenfeld klicken
- oder die folgenden AAO-Ereignisse in Skripten oder als Schaltflächenbelegung verwenden:
1 (>K:AAO_BUTTON_LAYER_UP/_DOWN/_TOGGLE), 0..3 (>K:AAO_BUTTON_LAYER_SET)
1 (>K:AAO_AXIS_LAYER_UP/_DOWN/_TOGGLE), 0..3 (>K:AAO_AXIS_LAYER_SET)
1 (>K:AAO_CONTROL_LAYER_TOGGLE), 0..n (>K:AAO_CONTROL_LAYER_SET)

Die UP- und DOWN-Ereignisse durchlaufen nur die aktiven Ebenen (= diejenigen, die Schaltflächen/Achsen enthalten), TOGGLE durchläuft alle vorhandenen Ebenen.

Sie können den Namen einer Ebene ändern, indem Sie mit der rechten Maustaste auf die Beschriftung klicken. Eine Ebene kann nur eine Beschriftung haben, und diese ist für Achsen und Schaltflächen identisch.

Die aktuelle Ebene kann über interne LVars abgefragt werden: (L:AaoButtonLayer) und (L:AaoAxisLayer).

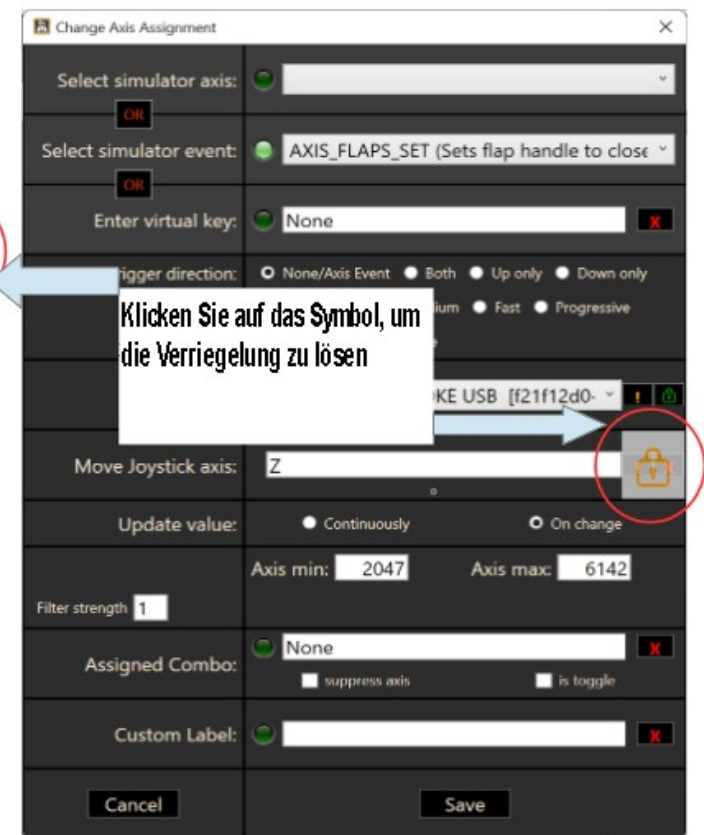
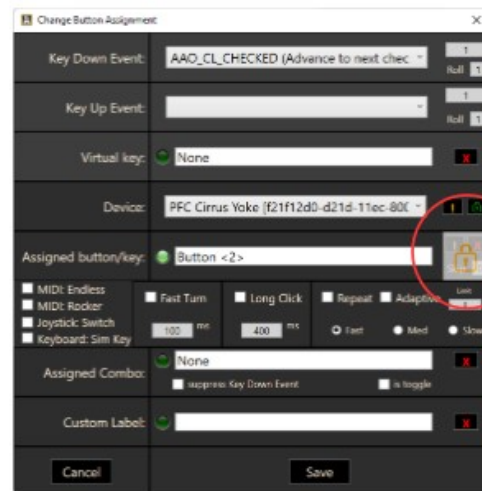
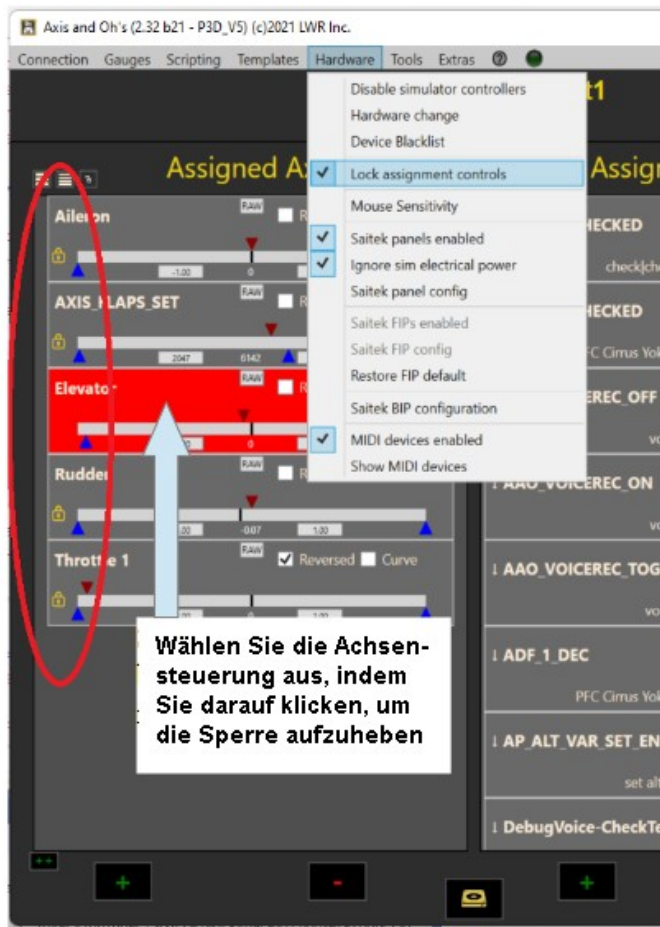
Die aktuellen Achsen- und Schaltflächenbeschriftungen können mithilfe der AAO-Ereignisse

1 (>K:AAO_EXPORT_BUTTON_LABELS) und 1 (>K:AAO_EXPORT_AXIS_LABELS) in LVars exportiert werden.

Die resultierenden LVars heißen (L:AAO_<AXIS|BUTTON>_LABEL_<layer>_<index>, String).

3.6 Sperren der GUI-Bedienelemente zur Vermeidung versehentlicher Änderungen

Die direkten Achsensteuerungen und die Dialoge „Achse hinzufügen/ändern“ bzw. „Schaltfläche hinzufügen/ändern“ können über „**Hardware** → **Zuordnungssteuerelemente sperren**“ gesperrt werden. Dies verhindert versehentliche Änderungen durch die Maus oder durch „störungsanfällige“ Controller-Hardware.



3.7 Verwaltung der Joystick-Achsenbelegung

3.7.1 Eine neue Achse zuweisen

The screenshot shows the 'Add Axis Assignment' dialog box with the following fields and options:

- Select simulator axis: [Dropdown menu]
- Select simulator event: [Dropdown menu]
- Enter Variable: [Text input] +
- Enter virtual event: [Text input: None] X
- Value range: Axis Min: -16384 Max: 16384 Shift: 0
- Update value: Continuously On change Rounding: Float Int
- Device: [Dropdown menu] !
- Move Joystick axis: [Text input: 0] ! X
- Assigned Combo: [Text input: None] X
 suppress axis is toggle
- Combo Variable: [Text input: L:] X
- Control Layer: [Dropdown menu: Default]
- Custom Label: [Text input] X
- Buttons: Cancel, Add

Der obere Bereich steuert die AUSGABE der Zuordnung.

Eingehende Ereignisse von deinem Eingabegerät werden in Simulatordaten umgewandelt und an den Simulator gesendet.

Du kannst entweder eine Achsenvariable oder ein Simulatorereignis auswählen, um Aktionen direkt im Simulator zu steuern, wie z. B. Querruder oder Gashebel. Nicht jede Achsenvariable oder jedes Ereignis funktioniert bei jedem Flugzeug; möglicherweise musst du einige Achsen und Ereignisse ausprobieren, bis du das richtige gefunden hast. Wenn du ein Ereignis auswählst, wird das Feld „Triggerrichtung“ sichtbar

Alternativ kannst du einen Variablennamen direkt eingeben oder dich dafür entscheiden ein virtuelles Ereignis zu senden.

Der untere Bereich zeigt die EINGABE der Zuweisung an.

Die Felder „Gerät“ und „Joystick-Achse“ sind zunächst leer und werden mit dem Eingabeereignis gefüllt, das die App empfängt, wenn du deinen externen Controller bewegst. Du kannst dieser Achse eine „Combo“-Taste zuweisen, sodass die Ausgabeaktion nur ausgelöst wird, wenn diese Taste gedrückt wird/wurde – oder nicht.

In den meisten Fällen bevorzugt MSFS für die primären Flugsteuerungen „Axis Events“ gegenüber „Variables“:
Die gängigsten Ereignisse wurden in der allerersten Gruppe der Ereignis-Baumstruktur dupliziert.

The 'Add Axis Assignment' dialog box is shown with the following fields and options:

- Select simulator axis: [Dropdown]
- Select simulator event: [Dropdown] (circled in red)
- Enter Variable: [Text] +
- Enter virtual event: [Text] X
- Value range: Axis min: -16384 Axis max: 16384
- Update value: Continuously On change Rounding: Float Int
- Trigger direction: None/Axis Event Both Up only Down only
- Repeat: Off Slow Medium Fast Progressive
- Device: T.16000M [4ca05ac0-0eaa-11ed-8002]
- Move Joystick axis: Slider 0
- Assigned Combo: [Dropdown] X
- Control Layer: Default
- Custom Label: [Text] X
- Buttons: Cancel, Add



The 'Select Simulator Event' dialog box displays a list of simulator events:

- !! Flight control axis
- AXIS_AILERONS_SET| Aileron
- AXIS_COLLECTIVE_SET| Collective
- AXIS_CYCLIC_LATERAL_SET| Cyclic Lateral
- AXIS_CYCLIC_LONGITUDINAL_SET| Cyclic Longitudinal
- AXIS_ELEV_TRIM_SET| Elevator Trim
- AXIS_ELEVATOR_SET| Elevator
- AXIS_FLAPS_SET| Flaps
- AXIS_HELICOPTER_THROTTLE_SET| Helicopter Throttle
- AXIS_HELICOPTER_THROTTLE1_SET| Helicopter Throttle 1
- AXIS_HELICOPTER_THROTTLE2_SET| Helicopter Throttle 2
- AXIS_LEFT_BRAKE_SET| Brake left
- AXIS_MIXTURE1_SET| Mixture 1
- AXIS_MIXTURE2_SET| Mixture 2
- AXIS_MIXTURE3_SET| Mixture 3
- AXIS_MIXTURE4_SET| Mixture 4
- AXIS_PROPELLER1_SET| Propeller 1
- AXIS_PROPELLER2_SET| Propeller 2
- AXIS_PROPELLER3_SET| Propeller 3
- AXIS_PROPELLER4_SET| Propeller 4
- AXIS_RIGHT_BRAKE_SET| Brake right
- AXIS_RUDDER_SET| Rudder
- AXIS_SPOILER_SET| Spoilers
- AXIS_STEERING_SET| Helicopter Steering
- AXIS_TAIL_ROTOR_SET| Tail Rotor
- AXIS_THROTTLE1_SET| Throttle 1
- AXIS_THROTTLE2_SET| Throttle 2
- AXIS_THROTTLE3_SET| Throttle 3
- AXIS_THROTTLE4_SET| Throttle 4
- AXIS_TILLER_SET| Tiller

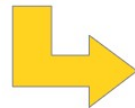
Buttons: Apply Filter, Clear Filter, Test with 0, Test with 1, Add custom event, Close

Ausgabe

The screenshot shows a configuration panel with several rows. Each row starts with a radio button and the text 'OR'. The rows are: 'Select simulator axis' (with a dropdown menu), 'Select simulator event' (with a dropdown menu), 'Enter Variable:' (with a text input field), and 'Enter virtual event:' (with a dropdown menu showing 'None'). Below these is a 'Value range' section with 'Axis Min: -16384', 'Max: 16384', and 'Shift: 0'. At the bottom is an 'Update value' section with radio buttons for 'Continuously' and 'On change', and a 'Rounding:' section with radio buttons for 'Float' and 'Int'. Colored circles highlight the dropdown menus (purple, green, red, blue) and the 'Update value' section (yellow).



Wählen Sie entweder
eine der vorkonfigurierten Achsvariablen
ODER
ein Ereignis vom Typ „Simulatorachse“
ODER
geben Sie einen Variablennamen ein
ODER
wählen Sie ein zu sendendes virtuelles Ereignis aus



Der Variablenname muss ohne Klammern eingegeben werden.
Beispiele:

L:simulatorlvarname, Number

A:AILERON POSITION, Position

L:aaolvarname

Hiermit wird festgelegt, wie die Achsposition an den Simulator gesendet wird:
Kontinuierlich (Continuously) = ständig
Bei Änderung (On change) = nur, wenn Sie den Hebel bewegen

Der Controller-Eingabewert wird auf den numerischen Bereich zwischen Min und Max abgebildet und kann nach oben oder unten verschoben werden. Die Wertfelder lassen sich mit dem Mausrad verändern. Wenn Sie darauf doppelklicken, können Sie den Wert über die Tastatur eingeben.

Der Achsausgabewert kann bei Bedarf auf den nächsten ganzzahligen Wert gerundet werden

Eingang

Device:	PFC Cirrus Yoke [f21f12d0-d21d-11ec	!	🔒
Move Joystick axis:	Z	!	X
Assigned Combo:	None	X	
	<input type="checkbox"/> suppress axis	<input type="checkbox"/> is toggle	
Custom Label:		X	



Bewegen Sie die gewünschte Joystick-Achse, bis sie in den Textfeldern angezeigt wird.

Sie können eine Achse oder den gesamten Controller mit den Ausrufezeichen **ignorieren** (Rechtsklick zum Zurücksetzen).



Sie können die Achse mit dem „**Sperr**“-Symbol an einen bestimmten Controller binden.

Bitte beachten Sie, dass das Kombinationsfeld „Gerät“ zunächst leer ist. Erst wenn Sie Ihre externe Achsensteuerung bewegen, wird es mit den Daten des eingehenden Ereignisses gefüllt.

Die Kästchen mit den Ausrufezeichen und dem Schloss können verwendet werden, um Geräte und Achsen zu isolieren oder zu ignorieren, falls diese die normale Erfassung stören. **Wenn Sie ein Gerät mit dem Schloss isolieren, wird der Eingang in den Modus „hohe Empfindlichkeit“ versetzt, wodurch Sie Steuerelemente erfassen können, die bei Bewegung nur sehr kleine Inkremente senden.**

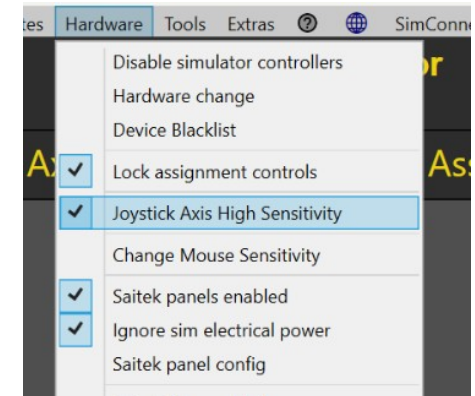
3.7.2 Empfindlichkeit der Joystick-Achsen

Im Normalbetrieb fragt AxisAndOhs die Joystick-Schnittstelle mit einer Frequenz von 18 Hz ab. Dabei wird nur die letzte vom Gamecontroller übermittelte Aktualisierung der Joystick-Achsenposition verarbeitet, alle anderen werden ignoriert. Dies gewährleistet einen schnellen Betrieb, ohne den Computer und die Schnittstelle zu sehr zu belasten.

Wenn Sie eine höhere Präzision benötigen, können Sie im Menü „Hardware“ den Modus „Joystick-Achsen – hohe Empfindlichkeit“ aktivieren.#

- In diesem Modus wird die Aktualisierungsrate auf 36 Hz erhöht.
- Alle vom Gamecontroller kommenden Aktualisierungen werden verarbeitet, (=nicht nur die letzte, sondern auch alle dazwischen)

Bitte beachten Sie, dass der Modus „Hohe Empfindlichkeit“ einen leistungsstarken Computer erfordert. Wenn Sie feststellen, dass die tatsächlichen Steuerungen im Simulator hinter Ihren Controller-Eingaben zurückbleiben, kann der Computer mit der hohen Aktualisierungsrate nicht Schritt halten.



Die hohe Empfindlichkeit kann jederzeit aktiviert oder deaktiviert werden. Du kannst auch die folgenden AAO-Ereignisse verwenden: (>K:AAO_HIGH_SENSITIVITY_ON), (>K:AAO_HIGH_SENSITIVITY_OFF), (>K:AAO_HIGH_SENSITIVITY_TOGGLE), die einer Taste oder einem Skript zugewiesen werden.

3.7.3 Achsen Kalibrieren

Jede zugewiesene Achse wird direkt im Hauptdialog kalibriert, indem Sie den Mauszeiger über die verschiedenen Steuerelemente bewegen und das Mausrad drehen.

The diagram shows a control interface for the 'Aileron' axis. It features a central slider with a grey area in the middle and darker areas on the sides. Above the slider are checkboxes for 'Reversed' and 'Curve'. Colored arrows point from text boxes to specific parts of the interface: a green arrow points to the left stop, a blue arrow to the center, a yellow arrow to the 'Reversed' checkbox, a red arrow to the 'Curve' checkbox, a purple arrow to the right stop, an orange arrow to the left edge, a dark blue arrow to the center of the slider, and a green arrow to the right edge.

Green box (top left): Klicken Sie auf das Dreieck, halten Sie die Maustaste gedrückt und bewegen Sie die Maus nach links/rechts, um größere Anpassungen vorzunehmen

Blue box (top middle): Drehen Sie hier das Mausrad, um den Mittelpunkt anzupassen

Yellow box (top right): Aktivieren, um die Achsenbewegung umzukehren

Red box (top far right): Aktivieren Sie die Funktion und drehen Sie das Mausrad, um die Reaktionskurve anzuwenden

Purple box (middle left): Klicken Sie mit der rechten Maustaste in den grauen Bereich, um an der aktuellen Position des Hebels eine Rastung zu erstellen.

Orange box (bottom left): Drehen Sie hier das Mausrad, um den linken Rand/die Totzone einzustellen

Dark blue box (bottom middle): Drehen Sie das Mausrad im grauen Bereich, um die mittlere Totzone einzustellen

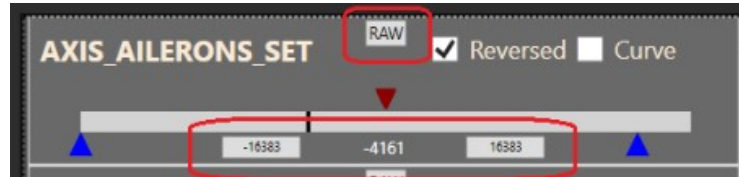
Green box (bottom right): Spin mouse wheel here to adjust right margin/deadzone

Dark purple box (middle left): Drehen Sie das Mausrad über einer Rastung, um deren Größe zu ändern.

Dark purple box (bottom left): Klicken Sie mit der rechten Maustaste auf eine Rastung, um sie zu entfernen.

3.7.4 Axis-Erweiterter Modus

Um präzise Änderungen an den Hauptachsenparametern vornehmen zu können, aktivieren Sie „Extras->Erweiterter Achsmodus“. Mit dieser Option zeigt die Achssteuerung den aktuellen Achswert an, der an die Simulation gesendet wird, sowie die aktuellen Minimal- und Maximalwerte der Achse und eine zusätzliche Schaltfläche „RAW“.

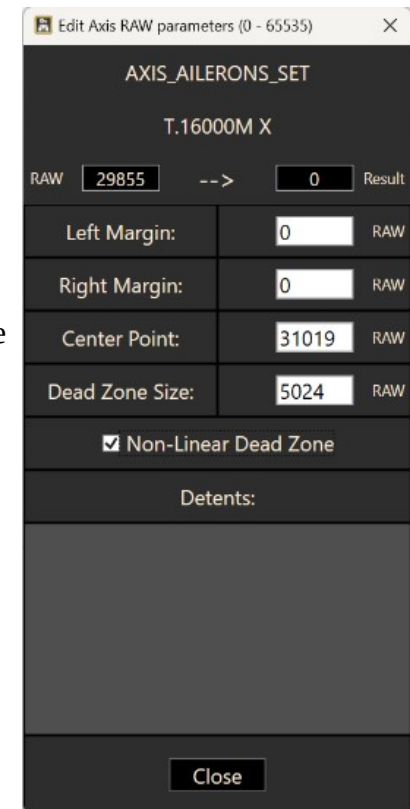


Um die Minimal- und Maximalwerte der Achse einzustellen, bewegen Sie die Achse, bis der gewünschte Wert unten (in der Mitte) angezeigt wird, und klicken Sie dann auf die Minimal- oder Maximalwerte links bzw. rechts. Um den Wert zurückzusetzen, klicken Sie mit der rechten Maustaste auf die Min- oder Max-Zahl.

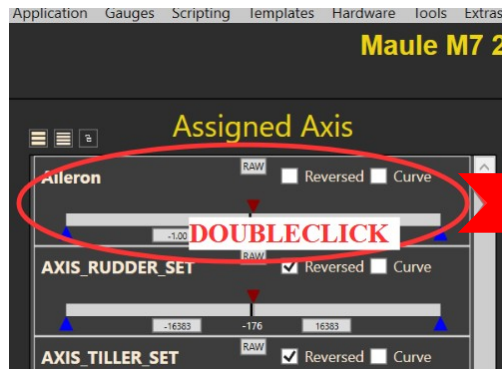
Mit der Schaltfläche „RAW“ können Sie einen Editor-Dialog für die Werte der Hauptachsenparameter aufrufen. Die weißen Zahlenfelder können mit dem Mausexplorer oder per Doppelklick geändert werden.

Die Option „Nichtlineare Totzone“ legt fest, ob die Achsenwerte ohne die Totzone auf den verbleibenden Bereich abgebildet werden oder ob die Totzone aus dem Bereich „herausgeschnitten“ wird.

Das „Ergebnis“ ist der Wert, der an die ausgewählte Variable oder das ausgewählte Ereignis im Simulator gesendet wird.



3.7.5 Eine Achsenzuordnung ändern



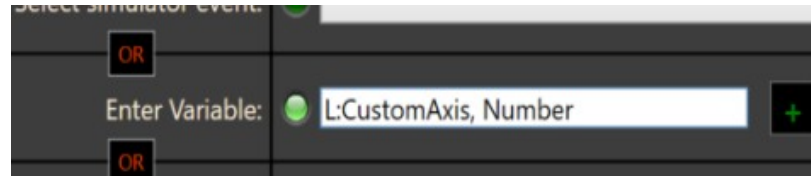
In diesem Dialogfeld können Sie die Zuordnung selbst sowie alle Achsenparameter ändern.

Die „Eingangsbereiche“ entsprechen den roten und blauen Dreiecken der Achsenvisualisierung.

Ein „Filter“-Wert über 1 erzwingt eine Mittelwertbildung auf der Achse, um „flatternde“ Potentiometer auszugleichen. Sie können zwischen einer progressiven und einer linearen Mittelwertbildung wählen.

Select Axis:	<input type="text"/>
OR	
Select Event:	<input type="text" value="AXIS_AILERONS_SET (Aileron)"/>
OR	
Enter Variable:	<input type="text"/>
OR	
Enter virtual event:	<input type="text" value="None"/>
<input type="checkbox"/> Axis Reversed	Axis Min: <input type="text" value="-16383"/> Max: <input type="text" value="16383"/> Shift: <input type="text" value="0"/>
Filter: <input type="text" value="1"/> <input type="radio"/> prog <input type="radio"/> linear	<input type="checkbox"/> Response Curve
Update value:	<input type="radio"/> Continuously <input type="radio"/> On change Rounding: <input type="radio"/> Float <input type="radio"/> Int
Trigger direction:	<input type="radio"/> None/Axis Event <input type="radio"/> Both <input type="radio"/> Up only <input type="radio"/> Down only
Repeat:	<input type="radio"/> Off <input type="radio"/> Slow <input type="radio"/> Medium <input type="radio"/> Fast <input type="radio"/> Progressive
Device:	<input type="text" value="T.16000M [4ca05ac0-0eaa-11ed-8002"/>
Move Joystick axis:	<input type="text" value="X"/>
Input ranges:	Left margin: <input type="text" value="0%"/> Center: <input type="text" value="50%"/> Right margin: <input type="text" value="0%"/> Deadzone: <input type="text" value="0%"/>
Combo Button:	<input type="text" value="None"/> <input type="checkbox"/> suppress axis <input type="checkbox"/> is toggle
Combo Variable:	<input type="text" value="L:"/>
Control Layer:	<input type="text" value="Default"/>
Custom Label:	<input type="text"/>
<input type="button" value="Cancel"/>	<input type="button" value="Save"/>

3.7.6 Einer Simulatorvariablen eine Achse zuweisen



Wenn Sie den Wert einer Simulatorvariablen über den zugewiesenen Regler steuern möchten, geben Sie den Typ, den Namen und die Einheit der Variablen in das Textfeld ein. Sie können Variablen über die Taste „+“ aus einer Liste auswählen, einige davon erfordern jedoch möglicherweise manuelle Korrekturen (wenn der Variablenname beispielsweise einen „:index“ enthält, muss dieser durch „:1“, „:2“ usw. ersetzt werden).

Achten Sie darauf, „Axis Min“ und „Axis Max“ an den für die Variable erforderlichen Wertebereich anzupassen.

3.7.7 Einmalige und virtuelle Ereignisse

Es ist auch möglich, einer Achse einmalige Ereignisse oder sogar virtuelle Tastenanschläge und vJoy-Ereignisse zuzuweisen. Je nach der Einstellung für die „Triggerrichtung“ löst die App das zugewiesene Ereignis oder die Taste aus, wenn Sie die Achse aus der Totzone in die zugewiesene Richtung bewegen (Beides, Nach oben, Nach unten) (Both, Up, Down), oder sendet den Wert kontinuierlich (Keine) (None).

Change Axis Assignment

Select Axis:

OR

Select Event:

OR

Enter Variable:

OR

Enter virtual event:

Trigger direction:

Repeating:

Mode:

Device:

Move Joystick axis:

Update value: Continuously On change Rounding: Float Int

Axis min: Axis max:

Filter strength

Assigned Combo: suppress axis is toggle

Custom Label:

Cancel Save

Klicken Sie hier, um das Zuweisungsfenster zu öffnen



Assign Virtual Event

Virtual keyboard key:

vJoy Button: Device # Button # OR POV # Direction

vJoy Axis: Device # Axis OR POV #

ViGEm X360 Button:

ViGEm X360 Axis: Axis OR Slider

Mouse action: Movement Scale

Mouse axis: Remember to assign a Combo!

Close

Geben Sie entweder eine Tastenkombination ein, indem Sie die gewünschten Tasten nacheinander drücken (=nicht gleichzeitig), oder wählen Sie eine vJoy-/ViGEm-/Maus-Aktion für den Trigger-Modus bzw. eine Achse für den kontinuierlichen Modus aus. Weitere Informationen finden Sie in den Kapiteln über die virtuellen Schnittstellen am Ende des Handbuchs.

Wenn Sie den Wert „Wiederholen“ auswählen, wiederholt die App den Tastendruck so lange, wie sich die Achse an der konfigurierten Position befindet. Wenn Sie „Progressiv“ auswählen, erhöht sich die Geschwindigkeit, mit der das Ereignis/die Taste wiederholt wird, sobald Sie den Hebel aus der Mittelposition bewegen.

Sie können auswählen, ob eine virtuelle Taste sofort ausgelöst und wieder losgelassen werden soll („Momentary“) oder ob jeder Tastendruck sie entweder aktiviert oder deaktiviert (Toggle).

3.7.8 Voreingestellte Kombination

Eine **Kombinations-Taste** (Combo Key) ist eine Joystick-/MIDI-Taste oder eine Tastenkombination auf der Tastatur, die zusätzlich zur zugewiesenen Achse gedrückt werden muss. Dies kann als Umschalter (erster Klick: Kombination EIN, zweiter Klick: Kombination AUS) oder im Dauerbetrieb (die zugewiesene Achse bewegt sich nur, solange die Kombinations-Taste gedrückt gehalten wird) verwendet werden. Klicken Sie auf die grüne LED, um das Eingabefeld zu aktivieren.

„**Achse/Ereignis unterdrücken**“ (Suppress Axis/Event): Wenn diese Option aktiviert ist, wird die ausgewählte Achse nur ausgelöst, wenn die Combo-Taste NICHT gedrückt ist. Auf diese Weise können Sie exklusive Umschalter erstellen.

Beispiel:

1. Eine Joystick-X-Achse ist der Steuerung des Querruders zugewiesen, ohne weitere Einstellungen
2. Dieselbe X-Achse ist der Steuerung des Seitenruders zugewiesen, mit der Joystick-Taste „5“ als Kombinations-Taste, „Achse unterdrücken“ ist NICHT aktiviert

In dieser Konfiguration bewegt die X-Achse das Querruder, wenn nichts gedrückt wird. Wenn „5“ gedrückt wird, bewegen sich Querruder UND Seitenruder.

3. Die Zuweisung des Querruders wird geändert: Die gleiche Joystick-Taste „5“ wird als Kombinations-Taste zugewiesen, jedoch mit aktivierter Option „Achse unterdrücken“.

Nun ist die Bewegung exklusiv: Wenn „5“ gedrückt wird, bewegt sich nur das Seitenruder; wenn die Taste losgelassen wird, bewegt sich das Querruder.

Sie können auch eine **Kombinationsvariable** (Combo Variable) zuweisen. Die Achse reagiert nur, wenn die zugewiesene Variable den Wert „1“ hat.

Die Variable hat immer den Typ „L:“

Beispiel: Textfeld enthält „MyVarName“ => die Variable heißt „(L:MyVarName)“

Achse deaktivieren:

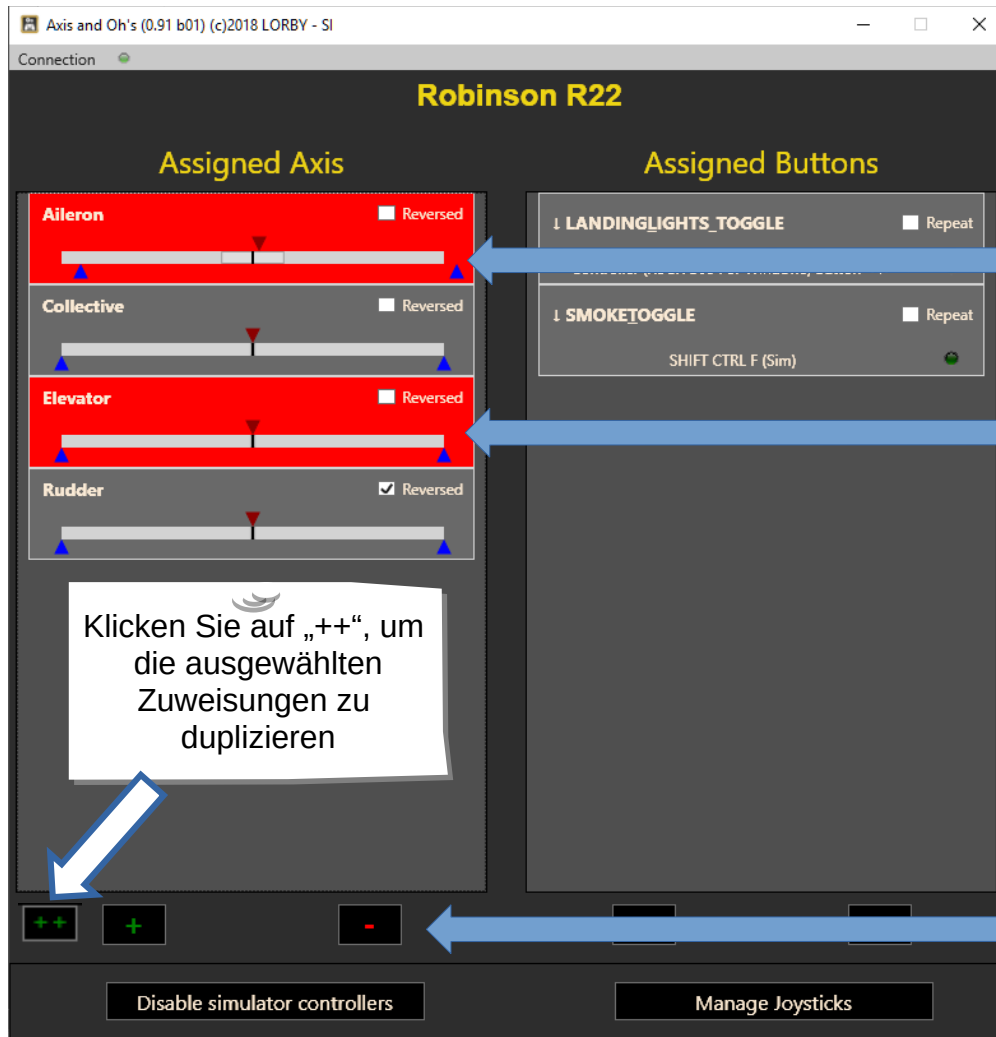
0 (>L:MyVarName)

Achse aktivieren:

1 (>L:MyVarName)

Solange die Variable nicht in AAO vorhanden ist, hat diese Einstellung keine Wirkung. Erst nachdem der LVar über ein Skript oder eine Schaltfläche ein Wert zugewiesen wurde, funktioniert die Achse entsprechend der Variable oder nicht.

3.7.9 Achse kopieren oder entfernen



Klicken Sie auf „++“, um die ausgewählten Zuweisungen zu duplizieren

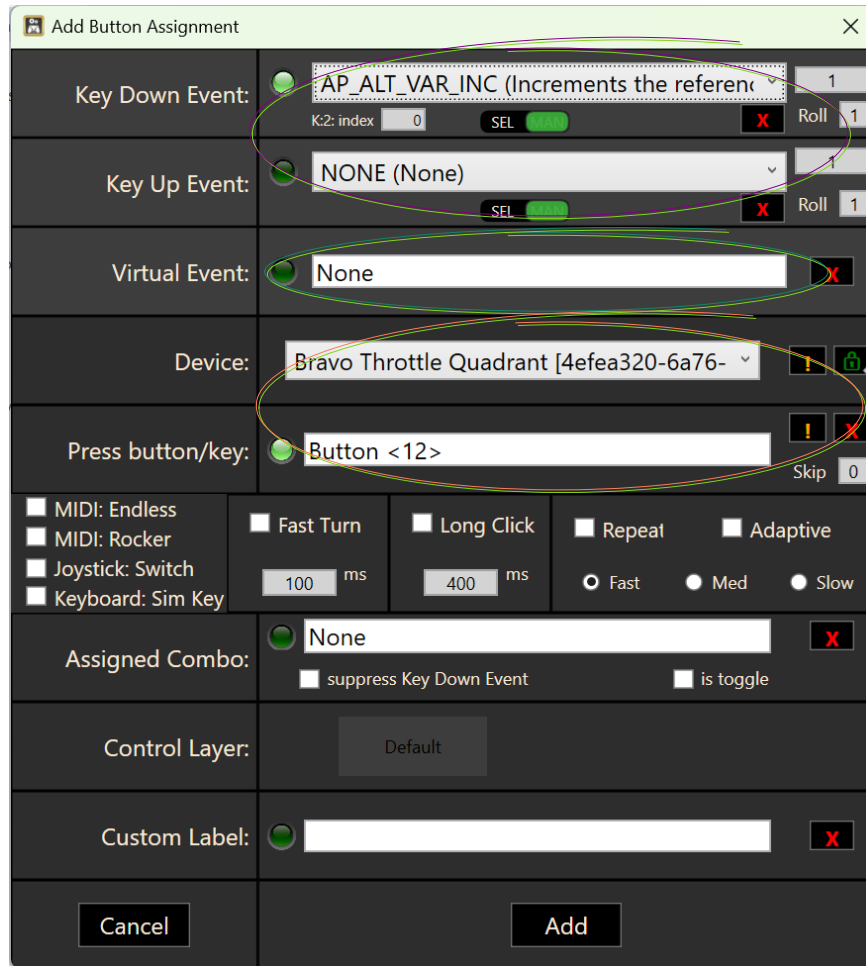
Message Box - Positivklicken Sie auf das Steuerelement, um es auszuwählen

Klicken Sie auf „-“, um die ausgewählten Steuerelemente zu entfernen

3.8 Verwaltung von Tasten-, MIDI- und Tastaturzuweisungen

3.8.1 Eine neue Taste zuweisen

Um Ihrem Flugzeug eine neue Taste zuzuweisen, klicken Sie auf das grüne „+“ unterhalb der Tastenliste auf der rechten Seite. Klicken Sie auf die grünen LEDs, um die zugehörigen Eingabefelder zu aktivieren.



Wählen Sie entweder

das Ereignis, das beim Drücken der Taste ausgelöst werden soll
UND/ODER
das Ereignis, das beim Loslassen der Taste ausgelöst werden soll

ODER

Weisen Sie eine virtuelle Tastatursequenz zu, die beim Drücken der Taste gesendet werden soll

Drücke die Taste auf deinem Joystick!

Du kannst eine Taste oder den gesamten Controller mit den Ausrufezeichen ignorieren (Rechtsklick zum Zurücksetzen)



Du kannst einen bestimmten Controller mit dem „Sperr“-Symbol sperren.

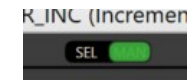
Für Tastatur-, Maus- oder Sprachzuweisungen muss das „Gerät“ auf diese Eingänge festgelegt sein.

Tastaturkombinationen müssen nacheinander eingegeben werden, drücke nicht alle Tasten gleichzeitig.

Beispiel: Für „Shift&Ctrl&F“ würdest du „Shift“ drücken, loslassen, dann „Ctrl“ drücken, loslassen und schließlich „F“ drücken

3.8.2 Auswahl vs. manuelle Zuweisung

Klicken Sie auf die grüne Schaltfläche unterhalb des Feldes zur Ereigniszuweisung, um zwischen der Ereignisauswahl und der manuellen Eingabe von Ereignissen zu wechseln.



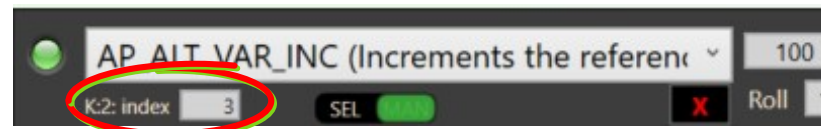
Im Auswahlmodus „SEL“ klicken Sie in das Ereignisfeld, um die Ereignisbaumliste zu öffnen. Wählen Sie ein Ereignis durch Doppelklick aus (siehe Kapitel 3.9 „Verwendung der Ereignisauswahlfelder“)

Im manuellen Modus „MAN“ können Sie jeden Ereignis- oder Variablennamen eingeben, den Sie an anderer Stelle übernommen haben. AAO muss diesen nicht im Voraus kennen, um ihn zu verwenden. Sie haben die Wahl zwischen der Eingabe von K:Events, LVars, BVars, HVars oder kleinen RPN-Skripten.

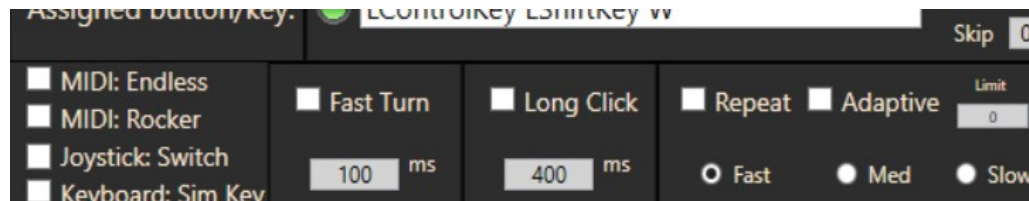
3.8.3 Der K:2:index

Einige der SDK-K:-Ereignisse benötigen einen zusätzlichen Parameter, den „Index“. Wenn man mit dem Mousrad einen Index von 1 oder höher einstellt, ergibt dies den RPN-Aufruf „Wert Index (>K:2:Ereignisname)“.

Beispiel: Alle AP-VAR-INC/DEC-Ereignisse benötigen einen Index; der MSFS-Autopilot verfügt in MSFS über mehrere „Slots“.
https://docs.flightsimulator.com/html/Programming_Tools/Event_IDS/Aircraft_Autopilot_Flight_Assist_Events.htm (flightsimulator.com) – siehe Spalte „Parameter“.



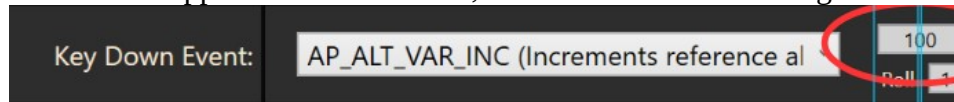
3.8.4 Parameter für die Buttonbetätigung



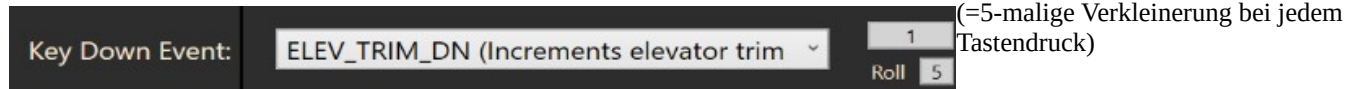
- MIDI: Endlos (Endless): für MIDI-Drehgeber mit Min/Max-Endlagen
- MIDI: Wippe (Rocker): für MIDI-Drehgeber, die nur drei Werte senden (links – Mitte – rechts)
- Joystick: Schalter (Switch): für Schalter (= keine Momenttasten) an Joystick-Geräten; dies synchronisiert die Schalterposition mit dem Simulator, wenn ein neues Flugzeug geladen wird.
- Tastatur (Keyboard): Sim-Taste: Verwenden Sie diese Option, wenn der Simulator Ihre Tastatur-Hotkeys „erfasst“
- Schnelle Drehung (Fast Turn): Diese Einstellung ist für Drehgeber gedacht. Wenn sie aktiviert ist, löst nur eine ausreichend schnelle Drehung das Key Down-Ereignis aus
- Langes Klicken (Long click): Das Key Down-Ereignis wird ausgelöst, wenn Sie die Taste länger als x ms gedrückt halten.
- Wiederholen (Repeat): Steuert, ob das Ereignis wiederholt werden soll, solange die Taste gedrückt gehalten wird, und legt die gewünschte Geschwindigkeit fest.
 - „Adaptiv“ beschleunigt die Geschwindigkeit, je länger die Taste gedrückt wird.
 - „Begrenzen“ (Limit) beendet die Wiederholung nach der angegebenen Anzahl von Betätigungen.
 - „Überspringen“ (Skip) macht das Gegenteil: Es ignoriert die von Ihnen festgelegte Anzahl von Eingabeereignissen, bevor das Key-Down-Ereignis erneut ausgelöst wird

3.8.5 Werte an Ereignisse senden

Bei einigen Ereignissen können Sie beim Drücken der Schaltfläche bestimmte Werte senden. Scrollen Sie mit dem Mausrad über die linke oder rechte Hälfte der numerischen Steuerelemente oder doppelklicken Sie darauf, um eine Zahl direkt einzugeben.

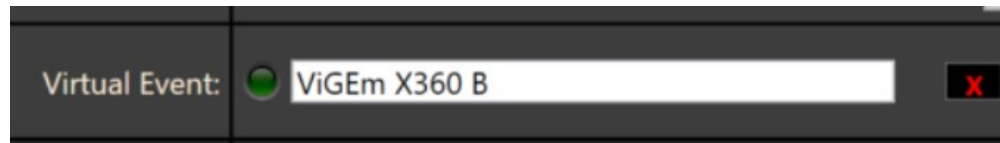


Mit der Steuerung „Roll“ wird das Ereignis so oft wiederholt, wie Sie es im Feld angeben:



3.8.6 Virtuelle Ereignisse

Wenn Sie dem Taster (Tastatur, vJoy, ViGEm, Maus) ein virtuelles Ereignis zuweisen, sendet die App diese Ereignisse beim Drücken des Tasters und simuliert so eine Eingabe dieser Geräte.



Um ein virtuelles Ereignis zuzuweisen, schalten Sie die grüne LED ein oder klicken Sie in das Textfeld.

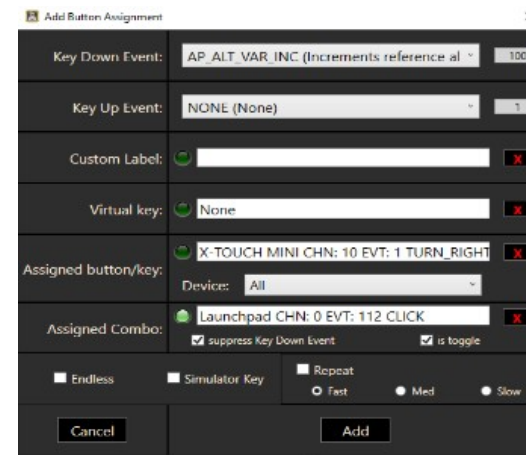
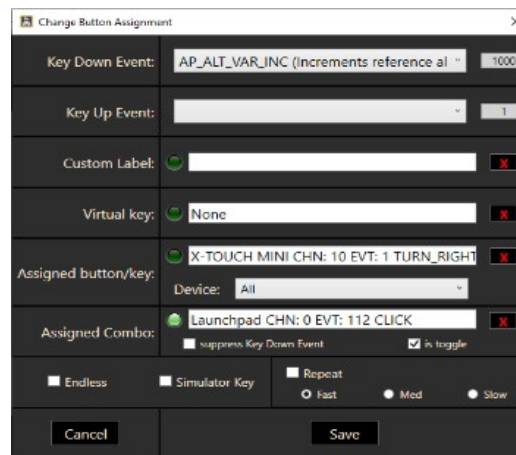
3.8.7 Voreingestellte Kombination



Eine Kombinations-Taste ist eine Joystick-/MIDI-Taste oder eine Tastenkombination auf der Tastatur, die zusätzlich zur zugewiesenen Taste gedrückt werden muss. Dies kann als Umschalter (erster Klick: Kombination EIN, zweiter Klick: Kombination AUS) oder im Dauerbetrieb (das Down-Ereignis wird nur ausgelöst, solange die Kombinations-Taste gedrückt gehalten wird) verwendet werden

„**Suppress Key Down Event**“: Wenn diese Option aktiviert ist, wird das ausgewählte Ereignis nur ausgelöst, wenn die Combo-Taste NICHT gedrückt ist. Auf diese Weise können Sie exklusive Umschalter erstellen.

Beispiel: Eine Rechtsdrehung des MIDI-Drehreglers soll zum Verringern der im AP ausgewählten Höhe verwendet werden. Sie würden zwei Zuweisungen für diesen Drehgeber mit derselben Kombinations-Taste erstellen. Eine Zuweisung setzt einen Wert von 100, wobei „Unterdrücken“ aktiviert ist, die andere setzt einen Wert von 1000, wobei „Unterdrücken“ NICHT aktiviert ist. In dieser Konfiguration verringert der Drehgeber die Höhe um 100 Fuß, wenn die Kombinations-Taste nicht gedrückt ist, und um 1000 Fuß, wenn die Kombinations-Taste gedrückt ist.



Um eine Kombination zuzuweisen, aktiviere die LED durch Anklicken. Deine Joystick-, MIDI- und Tastatureingaben werden dann an das Textfeld auf der rechten Seite weitergeleitet. Deaktiviere die LED, um zum normalen Betrieb zurückzukehren. Verwende „X“, um die Zuweisung aufzuheben und das Textfeld zu löschen. „**is toggle**“ schaltet die Toggle-Eigenschaft um, sodass du die Combo-Taste nicht ständig gedrückt halten musst.

Sie können auch eine **Combo-Variable** zuweisen. Die Taste wird nur ausgelöst, wenn die zugewiesene Variable den Wert „1“ hat.

Die Variable ist immer vom Typ „L:“

Beispiel: Textfeld enthält „MyVarName“ => die Variable heißt „(L:MyVarName)“

Achse deaktivieren:
0 (>L:MyVarName)

Achse aktivieren:
1 (>L:MyVarName)

Solange die Variable nicht in AAO existiert, hat diese Einstellung keine Wirkung. Erst nachdem der LVar über ein Skript oder eine Schaltfläche ein Wert zugewiesen wurde, funktioniert die Schaltfläche entsprechend der Variable.

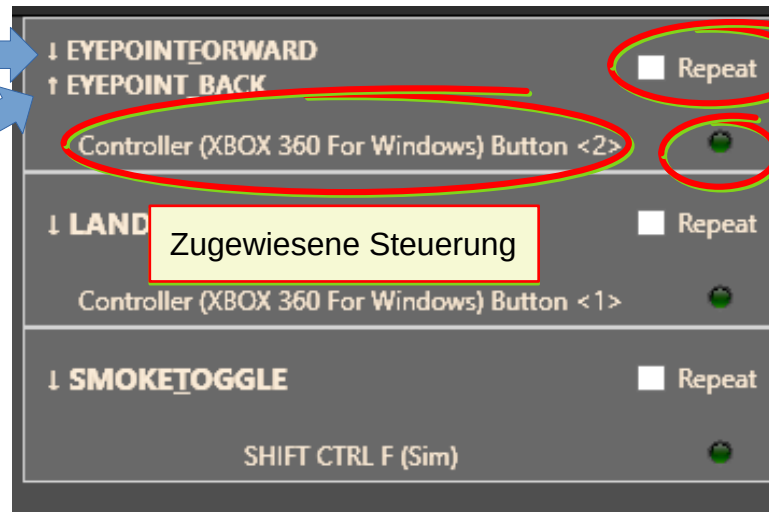
3.8.8 Langer Klick (Long click)

Wenn „Langer Klick“ aktiviert ist, muss die Taste mindestens so lange gedrückt gehalten werden, wie in Millisekunden angegeben, bevor das „Down“-Ereignis ausgelöst wird. Dem „Up“-Ereignis kannst du ebenfalls eine „Kurzdruk“-Aktion zuweisen, die sofort ausgelöst wird.

3.8.9 Die Steuerung der Tastenbelegung in der Hauptliste

Ereignis beim Drücken der Schaltfläche

Ereignis beim Loslassen der Schaltfläche

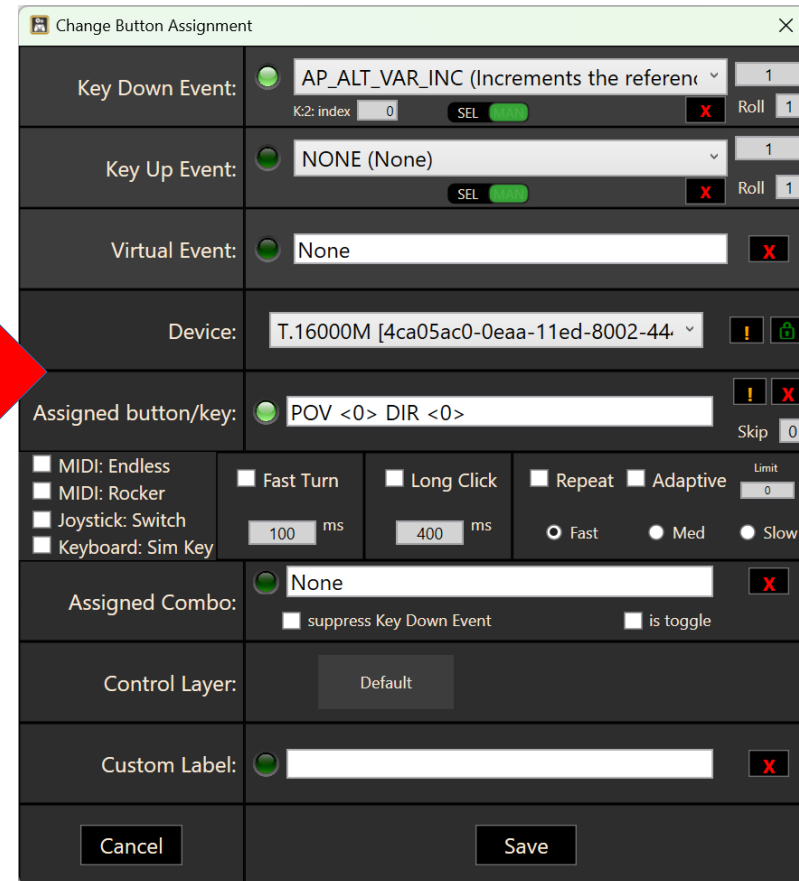
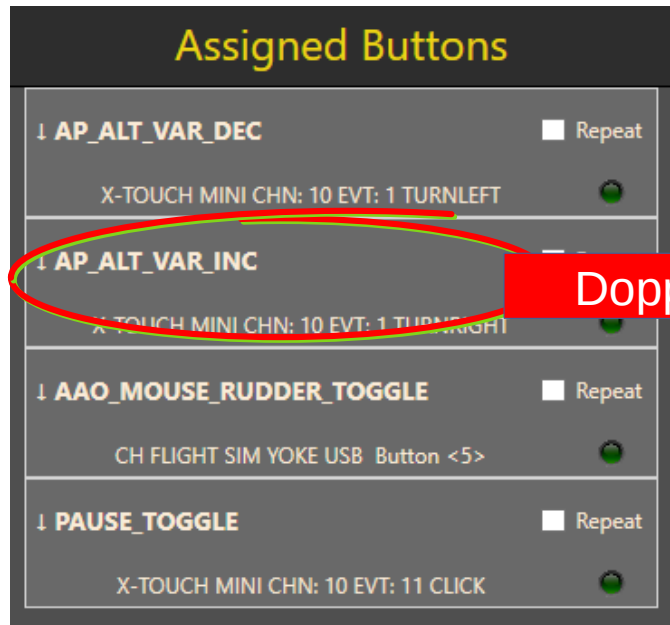


Einstellung wiederholen

Aktivitätsanzeige

Zugewiesene Steuerung

3.8.10 Tastenbelegung ändern



Das Dialogfeld „Tastenbelegung ändern“ funktioniert genauso wie das Dialogfeld „Hinzufügen“ (siehe Kapitel 3.4.1). Um eine bestimmte Taste in der Liste zu finden, wählen Sie „Extras → Taste suchen“ und betätigen Sie sie anschließend.

Klicken Sie auf „Speichern“, um diese Belegung zu speichern, oder auf „Abbrechen“, um sie zu verwerfen.

3.8.11 Zuweisungen für die Schaltflächen Kopieren oder Entfernen



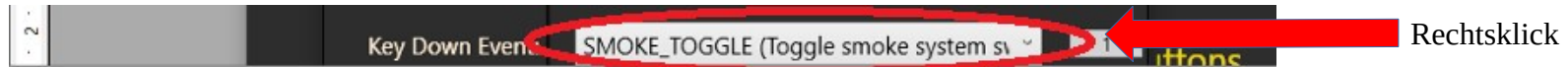
Klicken Sie auf die Steuerelemente, um sie auszuwählen

Klicken Sie auf „++“, um die ausgewählten Zuweisungen zu duplizieren

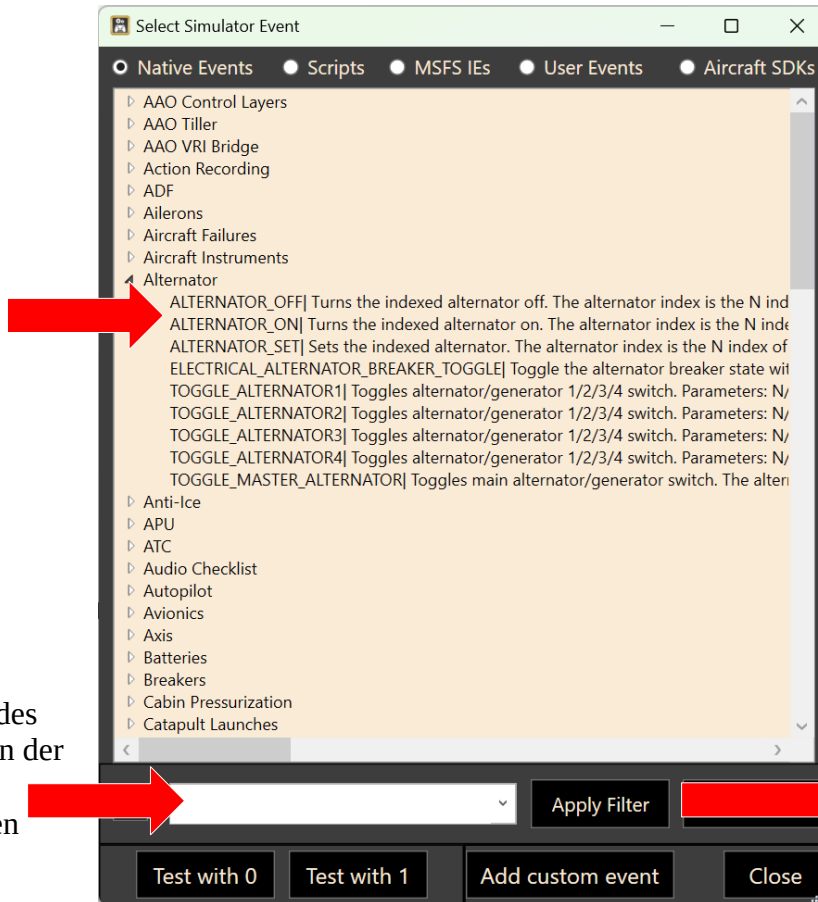
Klicken Sie auf „-“, um die ausgewählten Steuerelemente zu entfernen

3.9 Ereignisauswahlfelder verwenden

Ein Linksklick auf das Kombinationsfeld öffnet das Auswahldialogfeld, ein Rechtsklick setzt die Auswahl auf „Keine“ zurück



Doppelklicken Sie auf eine Ereignis-ID um sie auszuwählen



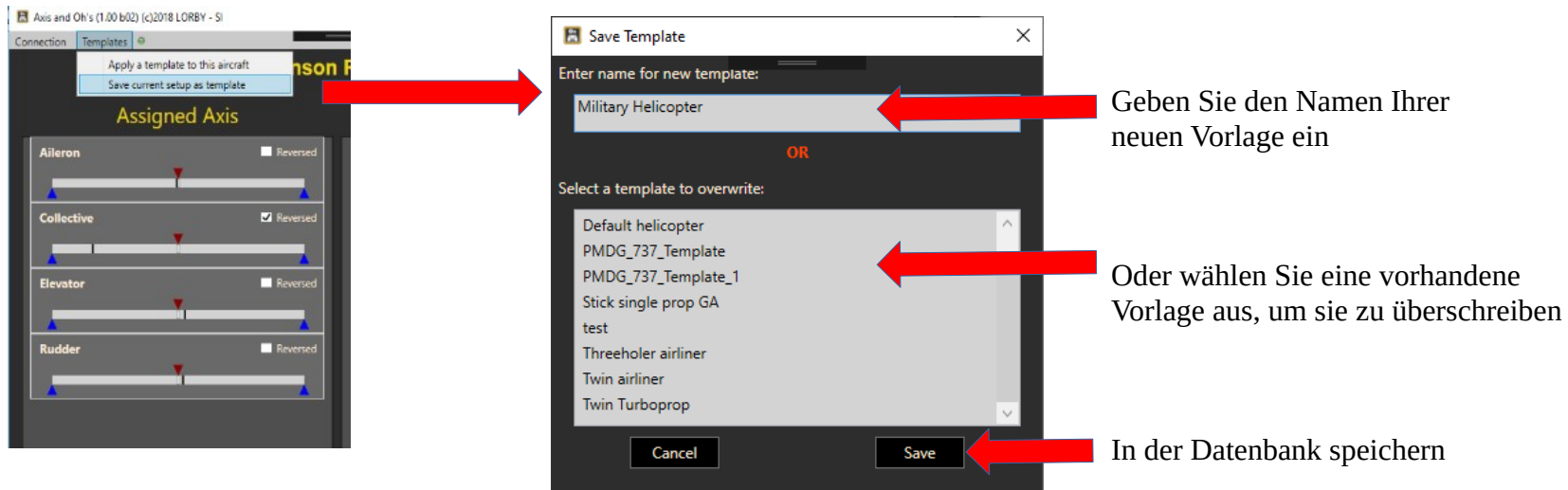
Filterbeispiel: Suche nach „trim“



3.10 Verwendung von Templates

Du kannst eine Joystick-Konfiguration als „Vorlage“ speichern, um sie schnell auf neue Flugzeuge anzuwenden. Vorlagen können in Dateien exportiert und wieder importiert werden. Wenn du eine Vorlage importierst, fordert dich die App auf, alle Steuerelemente in der Vorlage neu zu konfigurieren. Um ein Steuerelement zu ignorieren, klicke einfach auf „Abbrechen“, wenn der Zuweisungsdialog dafür erscheint.

Eine Konfiguration als Vorlage speichern

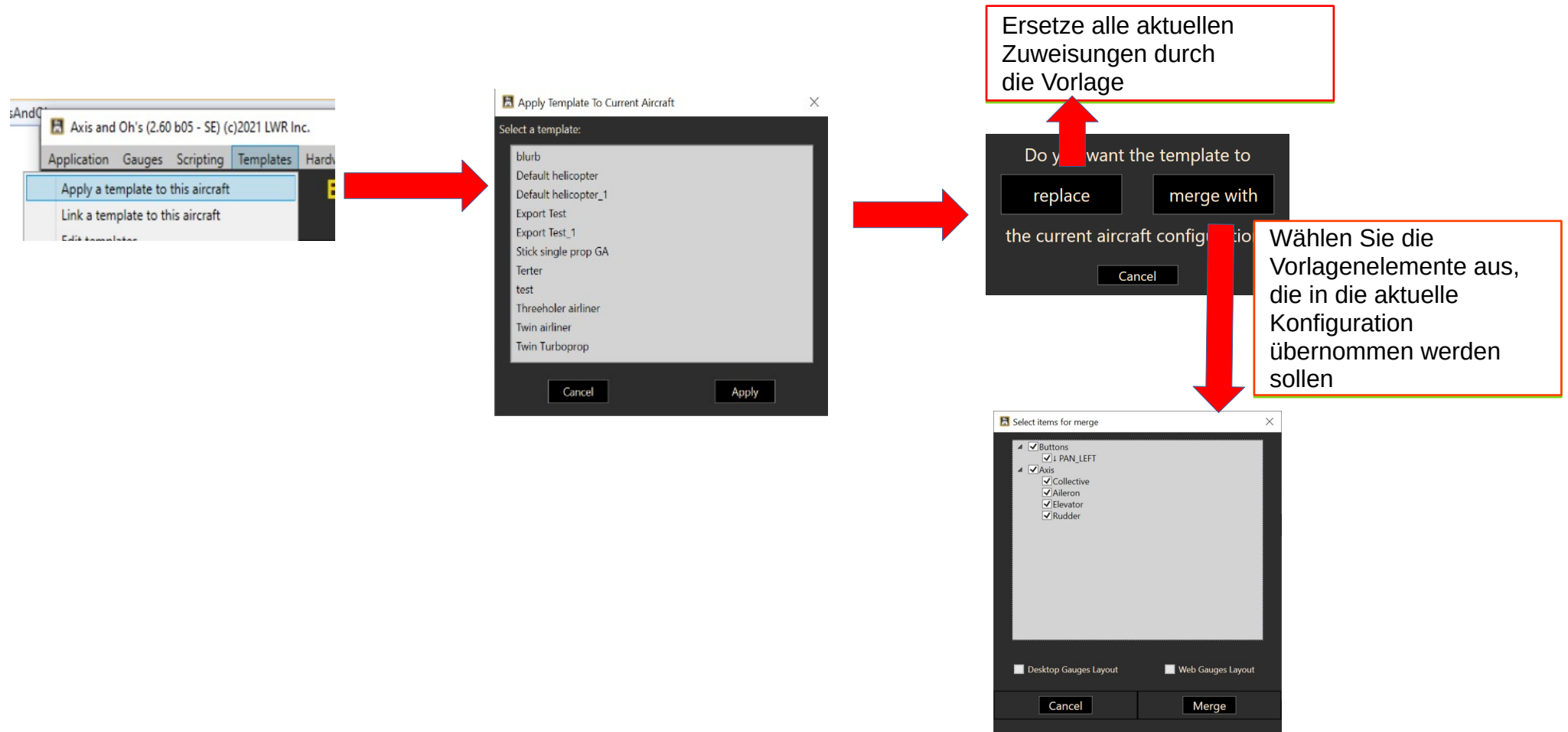


Ein Template ändern

Versetzen Sie das AAO in den Offline-Modus (rote LED) und laden Sie die Vorlage wie eine Konfiguration. Die Vorlagen befinden sich am Ende der Auswahlliste.

Eine Vorlage auf Ihr aktuelles Flugzeug anwenden

Sobald Ihr Flugzeug erkannt wurde, wählen Sie im oberen Menü „Vorlagen → Vorlage auf dieses Flugzeug anwenden“. Durch das Anwenden einer Vorlage werden Kopien der Schaltflächen und Achsen der Vorlage in der aktuellen Konfiguration erstellt.



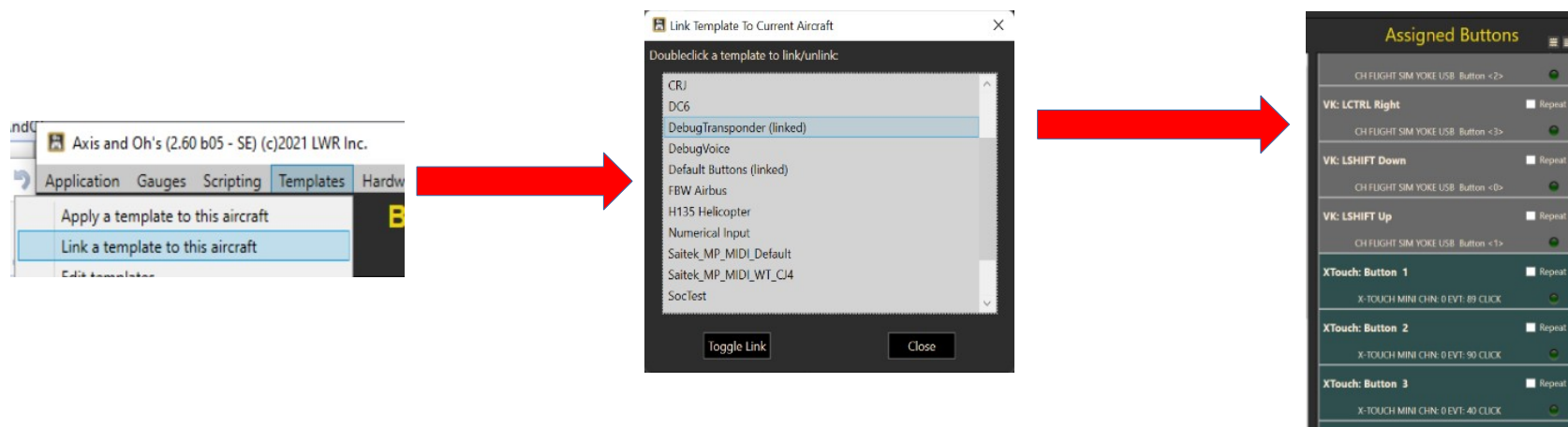
Eine Änderung der verwendeten Vorlage führt NICHT zu einer Aktualisierung der damit erstellten Flugzeugkonfigurationen.

Eine Vorlage mit Ihrem aktuellen Flugzeug verknüpfen

Anstatt eine Vorlage anzuwenden, können Sie diese über „Vorlagen → Vorlage mit diesem Flugzeug verknüpfen“ mit Ihrem aktuellen Flugzeug verknüpfen.

In diesem Fall erstellt AAO keine Kopie der Tasten und Achsen in Ihrer Flugzeugkonfiguration, sondern verweist lediglich auf die Vorlage. Eine Änderung der Vorlage wirkt sich direkt auf alle verknüpften Flugzeugkonfigurationen aus.

Die Tasten und Achsen einer verknüpften Vorlage werden in einer anderen Farbe angezeigt.

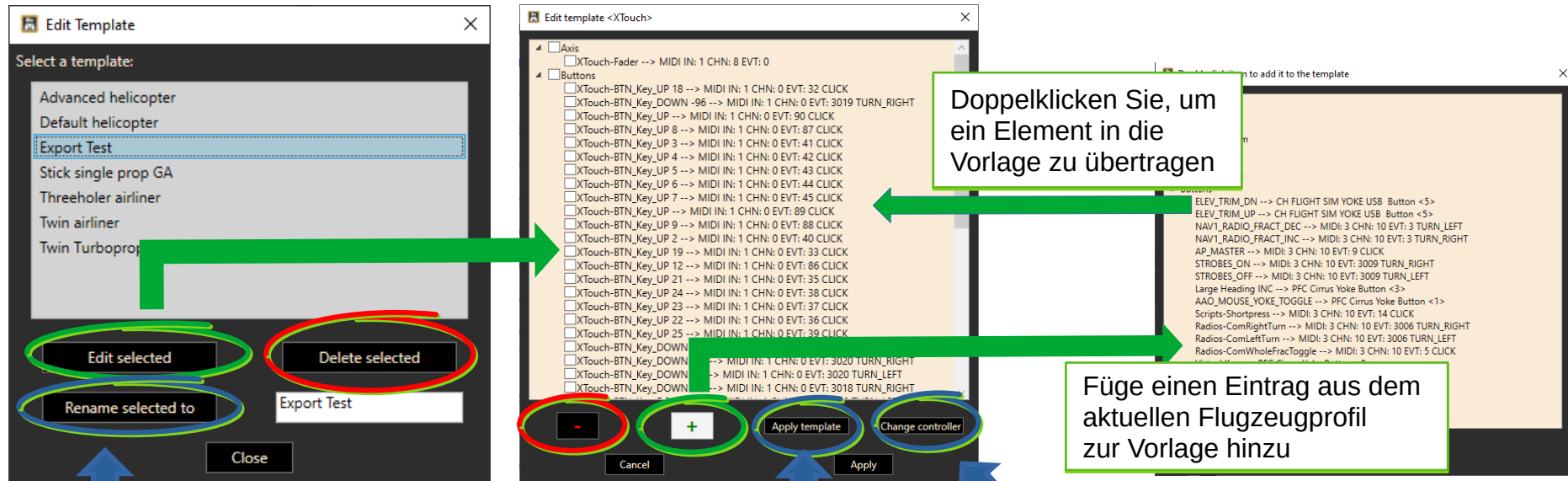


Um eine Vorlage zu verknüpfen, doppelklicken Sie in der Liste darauf. Vorlagen, die bereits mit Ihrem Flugzeug verknüpft sind, werden als „(verknüpft)“ angezeigt – doppelklicken Sie darauf, wenn Sie die Verknüpfung aufheben möchten.

Eine Änderung an der verknüpften Vorlage WIRKT SICH auf alle zugehörigen Flugzeugkonfigurationen aus.

Vorlagen und Konfigurationen bearbeiten

Sie können vorhandene Vorlagen und Konfigurationen unter „Vorlagen -> Vorlagen bearbeiten / Konfigurationen bearbeiten“ bearbeiten.



Benennen Sie die Vorlage in den Inhalt des Textfelds um.

Die ausgewählte Vorlage löschen

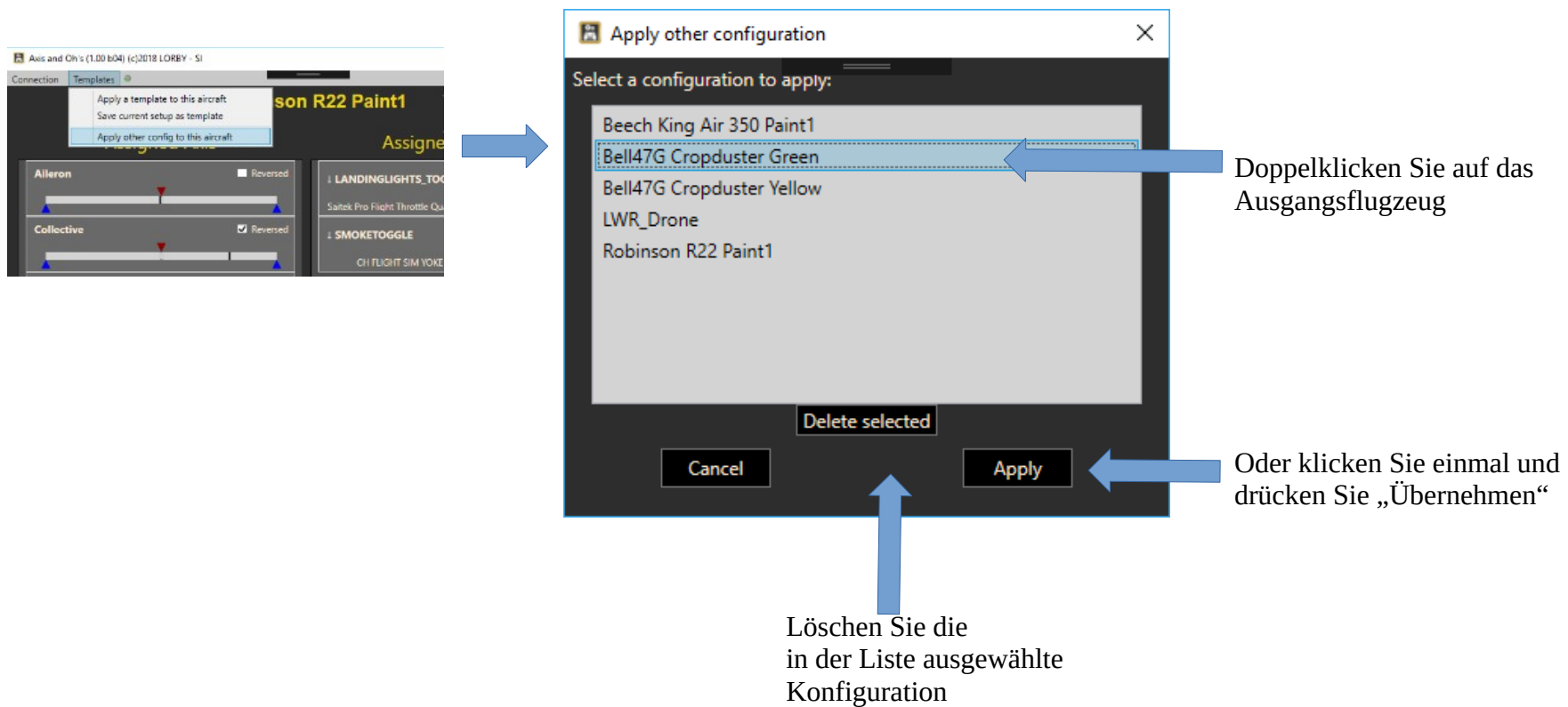
Die ausgewählten Elemente löschen

Eine weitere Vorlage hinzufügen

Ändern Sie die Controller-Zuordnung für die ausgewählten Elemente

Eine vorhandene Konfiguration auf Ihr aktuelles Flugzeug anwenden

Sobald Ihr Flugzeug erkannt wurde, wählen Sie im oberen Menü „Vorlagen → Andere Konfiguration auf dieses Flugzeug anwenden“



The image shows a screenshot of the flight simulator interface with a dialog box titled "Apply other configuration" open. The dialog box contains a list of configurations to apply to the current aircraft. The configurations listed are:

- Beech King Air 350 Paint1
- Bell47G Cropduster Green
- Bell47G Cropduster Yellow
- LWR_Drone
- Robinson R22 Paint1

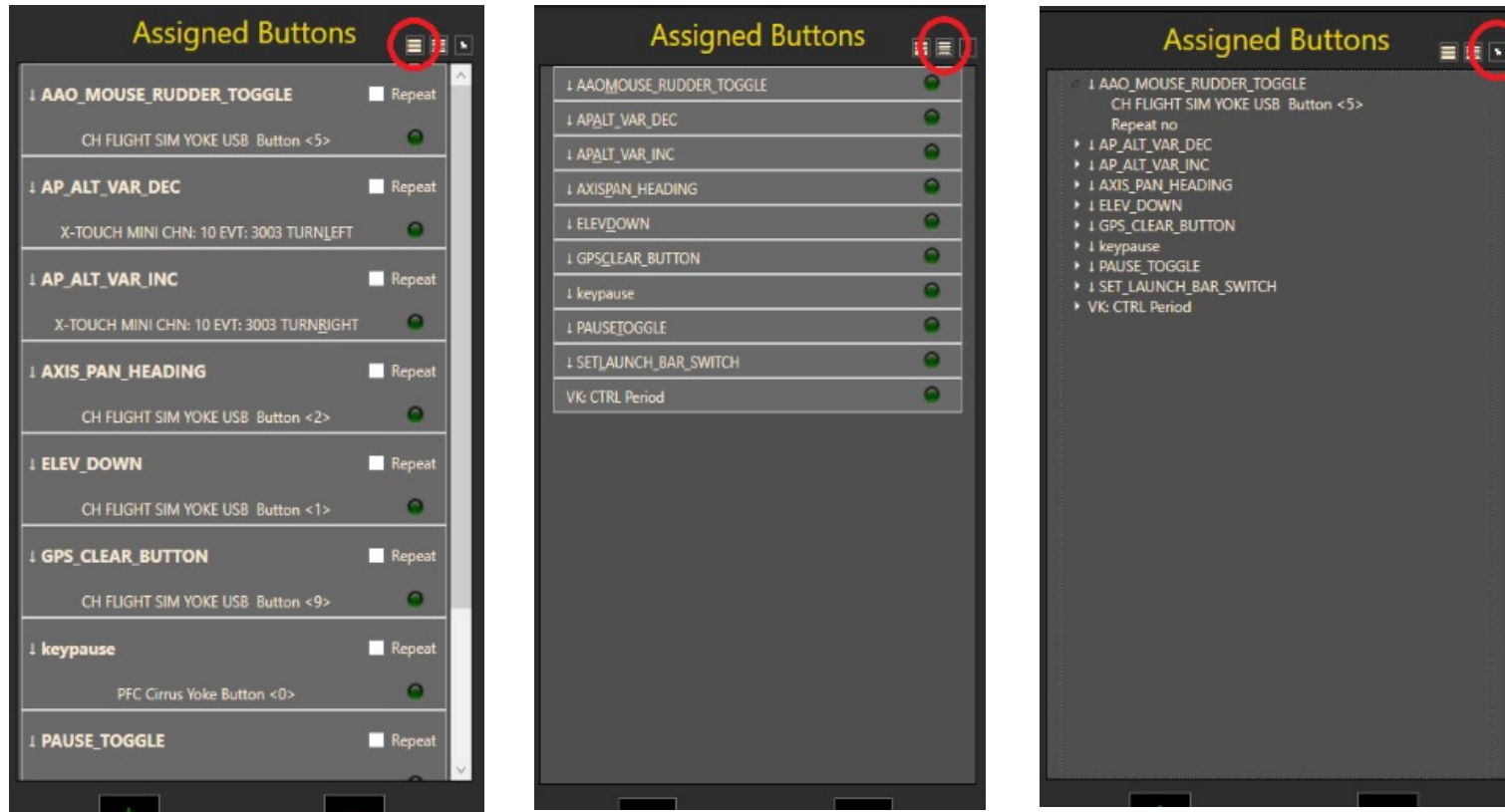
The "Bell47G Cropduster Green" configuration is selected, indicated by a blue dashed border. Below the list are three buttons: "Cancel", "Delete selected", and "Apply".

Annotations with blue arrows provide instructions:

- An arrow points to the "Apply" button with the text: "Oder klicken Sie einmal und drücken Sie „Übernehmen“"
- An arrow points to the "Delete selected" button with the text: "Löschen Sie die in der Liste ausgewählte Konfiguration"
- An arrow points to the selected configuration in the list with the text: "Doppelklicken Sie auf das Ausgangsflugzeug"

3.11 Ansichtsmodi des Bedienfelds

Die Achsen- und Schaltflächenleisten lassen sich über die Symbole oben in drei verschiedene Ansichtsmodi umschalten



Die Baumansicht bietet eine besondere Anwendungsmöglichkeit: Wenn Sie einer Schaltfläche eine benutzerdefinierte Bezeichnung zuweisen, können Sie diese verwenden, um die Schaltflächen in der Baumansicht zu gruppieren. Dazu muss die Bezeichnung im Format „Gruppenname:Bezeichnungname“ angegeben werden.

4 Spracherkennung

AAO kann so konfiguriert werden, dass es die in Ihrem Windows-System integrierte Standard-Spracherkennung nutzt. Indem Sie im Dialogfeld zur Tastenbelegung Sprachbefehle hinzufügen, können Sie Ereignisse und Skripte auslösen, indem Sie den Befehl in das Mikrofon Ihres Standard-Audiogeräts sprechen. Es steht eine „Push-to-Talk“-Option zur Verfügung, falls Sie die Spracherkennung nur bei Bedarf manuell aktivieren möchten.

Bitte beachten Sie:

- Die Spracherkennung ist nur so gut wie Ihr lokales Windows-System. Sie müssen sie ordnungsgemäß trainieren, damit sie Ihre Phrasen zuverlässig erkennt.
- Die Windows-Spracherkennung ist lokalisiert. Wenn Sie verschiedene Sprachen verwenden möchten, müssen Sie Ihrem Windows-System die entsprechenden Microsoft-Sprachpakete hinzufügen.
- Sie können im Menü „**Extras->Sprache für Spracherkennung auswählen**“ (Extras->Select language for speech recognizer) eine bestimmte Sprache für die Spracherkennung auswählen. Doppelklicken Sie auf eine Sprache in der Liste, um sie auszuwählen. Verwenden Sie das rote „X“, um die Auswahl auf Ihre Windows-Standard-Einstellung zurückzusetzen. Dies kann erforderlich sein, wenn sich Ihre Standard-Spracherkennung von Ihrer Standard-Systemsprache unterscheidet.
(Hinweis: Dieser Dialog trennt AAO vom Simulator)

Einem Ereignis eine Sprachphrase zuweisen

Um eine Sprachbefehlsphrase zuzuweisen, verwenden Sie die Dialoge der Schaltfläche „Hinzufügen/Ändern“:

- Wählen Sie zunächst „Voice“ aus dem Steuerelement „Device“ aus
- Geben Sie dann die gewünschte Phrase in das Textfeld „Assigned Button/Key“ ein
- Sie können derselben Zuweisung mehrere Phrasen hinzufügen, indem Sie das | „Pipe“-Symbol verwenden: „alle Lichter|volle Beleuchtung|Lichter umschalten“
- Sie können AAO anweisen, zu warten, bis Sie die PTT-Taste loslassen, bevor das Ereignis ausgelöst wird, indem Sie am Ende der Phrase ein „!“ hinzufügen. Dies ist für die Erfassung numerischer Eingaben erforderlich.

Change Button Assignment

Key Down Event: ALL_LIGHTS_TOGGLE (Toggle all lights) Roll 1

Key Up Event: Roll 1

Custom Label: Select "Voice" here

Device: Voice

Virtual key: None Type in a phrase using your keyboard

Assigned Button/Key: all lights

Assigned Combo: None suppress Key Down Event is toggle

Endless Fast Turn Long Click Repeat Adaptive
Sim Key 100 ms 400 ms Fast Med Slow

Cancel Save

Die SRGS-Grammatik

Anstelle von durch das Pipe-Zeichen (|) getrennten Phrasen können Sie auch den Namen einer XML-Datei im SRGS-Format angeben. Geben Sie entweder einen absoluten Pfad (C:\myfolder\...\agrammar.xml) oder einen relativen Pfad zu „\Documents\LorbyAxisAndOhs Files\Grammars\“ ein (also einfach „agrammar.xml“, wenn sich diese Datei im Ordner \Grammars\ befindet).

SRGS-Spezifikation:

[Speech Recognition Grammar Specification Version 1.0 \(w3.org\)](http://www.w3.org/Speech-Recognition-Grammar-Specification-Version-1.0)

Beispiel:

```
<?xml version="1.0" encoding="UTF-8" ?>
<grammar version="1.0" xml:lang="en-US"
  xmlns="http://www.w3.org/2001/06/grammar" tag-format="semantics/1.0" root="Main">
  <!-- Defines an SRGS grammar for requesting a flight. This grammar includes a Cities rule that lists
  the cities that can be used for departures and destinations.
  Definiert eine SRGS-Grammatik für die Fluganfrage. Diese Grammatik enthält eine „Cities“-Regel, in
  der die Städte aufgeführt sind, die als Abflug- und Zielorte verwendet werden können.-->
  <rule id="Main">
    <item> I would like to fly from <ruleref uri="#Cities"/> to <ruleref uri="#Cities"/>
    </item>
  </rule>
  <rule id="Cities" scope="public">
    <one-of>
      <item> Seattle </item>
      <item> Los Angeles </item>
      <item> New York </item>
      <item> Miami </item>
    </one-of>
  </rule>
</grammar>
```

Zahleneingabe

Sie können Phrasen einrichten, bei denen Zahlen eingegeben werden können. Alle Ziffern müssen einzeln ausgesprochen werden, gefolgt von einem Multiplikator von 100 oder 1000. Mit „Zurück“ wird die letzte Ziffer gelöscht. Die Grammatik für die Zahleneingabe kann über den Dialog

„Extras->Numerische Mustererkennung“

angepasst werden. In diesem Dialog können Sie die erkannten Phrasen eingeben. Verwenden Sie das Pipe-Symbol „|“, um mehrere Optionen derselben Phrase hinzuzufügen.

Beispiel: Einstellen der AP-Höhe

1. Stellen Sie sicher, dass Sie eine PTT-Taste zuweisen (siehe nächstes Kapitel)
2. Erstellen Sie ein Skript:
Skriptgruppe: Skripte
Skriptname: VoiceApAlt
Skriptcode: (L:VoiceRecNum) (>K:AP_ALT_VAR_SET_ENGLISH)
3. Erstellen Sie eine Sprachzuweisung
Tastenbetätigungsereignis: „Scripts-VoiceApAlt“
Phrase: „set altitude!“

Das ! ist obligatorisch; es weist AAO an, zu warten, bis die PTT-Taste losgelassen wurde, damit Sie die Zahlen sprechen können.

Aktion: Drücken Sie die PTT-Taste, sprechen Sie „set altitude two one thousand“ (Höhe auf 2.000 einstellen), lassen Sie die PTT-Taste los.

Sprechen Sie langsam – lassen Sie eine kurze Pause zwischen dem Text und den Zahlen.

Number	Phrase
0:	zero null
1:	one eins
2:	two zwei
3:	three drei
4:	four vier
5:	five fünf
6:	six sechs
7:	seven sieben
8:	eight acht
9:	nine neun
Decimal:	decimal komma
100:	hundred hundert
1000:	thousand tausend
back:	back zurück

Close

Verwendung der numerischen Eingabe

Es gibt zwei Möglichkeiten, die numerische Eingabe zu nutzen:

1. Einfache Ganzzahlwerte, wie Höhenangaben oder Gradzahlen, können Sie direkt in der Aufgabe mit Ereignissen verwenden, die Eingabewerte verarbeiten.

Beispiel: Einstellen der AP-Höhe

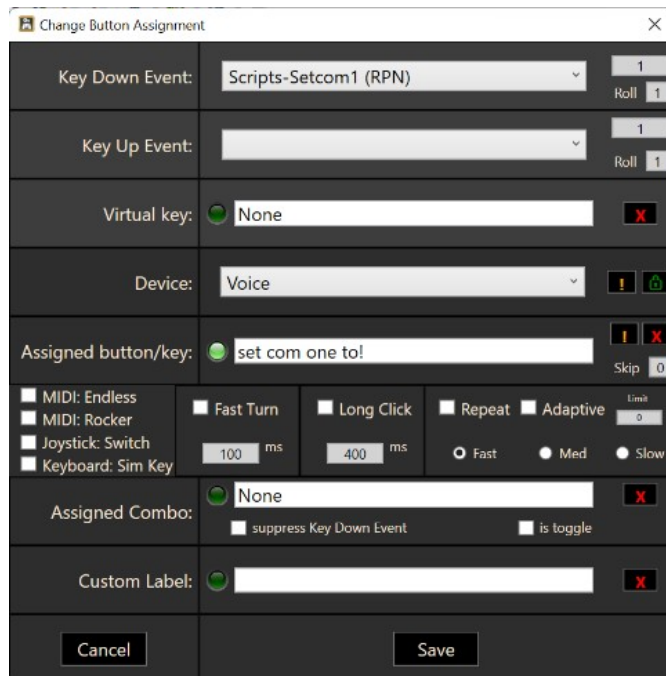
The screenshot shows the 'Change Button Assignment' dialog box. The 'Key Down Event' is set to 'AP_ALT_VAR_SET_ENGLISH (Sets altitude r...' with a value of '0' and a 'Roll' of '1'. The 'Key Up Event' is empty with a 'Roll' of '1'. The 'Virtual key' is set to 'None'. The 'Device' is set to 'Voice'. The 'Assigned button/key' is set to 'set altitude!' with a 'Skip' of '0'. The 'Assigned Combo' is set to 'None'. The 'Custom Label' is empty. The dialog has 'Cancel' and 'Save' buttons at the bottom.

Wenn eine numerische Eingabe eingeht, ignoriert AAO das Feld „Key Down Event Value“ und verwendet stattdessen die Spracheingabe.

In diesem Beispiel würden Sie „set altitude two two thousand“ sagen, um eine AP-Höhe von 22.000 Fuß auszuwählen.

2. Zahlen mit Dezimalstellen, wie beispielsweise Frequenzen oder komplexere Zusammenhänge, können mit RPN-Skripten verarbeitet werden. Der erkannte Zahlenwert steht als lokale Lvar (L:AoVoiceRecNum) zur Verfügung.

Beispiel: Einstellen der Frequenz für COM 1



Der Sprachbefehl würde lauten: „Set Com One to One Two Four Decimal Two Five“ (Setze com auf eins, eins, zwei, vier, zwei Komma fünf), um eine Frequenz von 124,25 einzustellen.

NATO-Alphabet

Ähnlich wie bei der numerischen Eingabe können Sie Zeichenfolgen mithilfe des NATO-Alphabets eingeben.

„Extras->NATO alphabet recognition“ (Extras->NATO-Alphabet-Erkennung)

In diesem Dialogfeld können Sie die erkannten Phrasen eingeben. Verwenden Sie das Pipe-Zeichen „|“, um mehrere Optionen derselben Phrase hinzuzufügen.

Um die NATO-Alphabet-Eingabe zu nutzen, fügen Sie Ihrer Erkennungszeichenfolge ein „#“ hinzu.

Sie können die Kennzeichen „!“ und „#“ nicht gleichzeitig verwenden.

Die erkannten Buchstaben werden unter „(L:AaoVoiceRecNato, String)“ gespeichert.

NATO alphabet recognition phrases			
A:	alfa	S:	sierra
B:	bravo	T:	tango
C:	charlie	U:	uniform
D:	delta	V:	victor
E:	echo	W:	whiskey
F:	foxtrot	X:	xray x-ray
G:	golf	Y:	yankee
H:	hotel	Z:	zulu
I:	india	0:	zero
J:	juliett	1:	one
K:	kilo	2:	two
L:	lima	3:	three
M:	mike	4:	four
N:	november	5:	five
O:	oscar	6:	six
P:	papa	7:	seven
Q:	quebec	8:	eight
R:	romeo	9:	nine
		Back:	back
Default		Close	

Hinzufügen der PTT-Taste

Wenn Sie nicht möchten, dass AAO ständig auf Empfang ist, können Sie eine PTT-Taste wie folgt implementieren:

- Die erforderlichen Ereignisse finden Sie in der Ereignis-Baumstruktur in der Gruppe „Spracherkennung“
- Wählen Sie „AAO_VOICEREC_ON“ als Down-Ereignis und „AAO_VOICEREC_OFF“ als Up-Ereignis

ODER

- Wählen Sie „AAO_VOICEREC_TOGGLE“ als Down-Event aus

Change Button Assignment

Key Down Event: AAO_VOICEREC_ON (PTT - voice recognit... Roll 1

Key Up Event: AAO_VOICEREC_OFF (PTT - voice recognit... Roll 1

Custom Label: [Empty field] X

Device: PFC Cirrus Yoke [1e6c10b0-8f01-11eb-80... ! [Green indicator]

Virtual key: None X

Assigned button/key: Button <6> [Lock icon]

Assigned Combo: None X
 suppress Key Down Event is toggle

Endless Fast Turn Long Click Repeat Adaptive
 Sim Key 100 ms 400 ms Fast Med Slow

Cancel Save

Alle „AAO_VOICEREC_xx“-Ereignisse können auch Sprachbefehlen zugewiesen werden; sie selbst sind unabhängig von der PTT-Logik.

Vom Spracherkennungsmodul verwendete Lvars

(L:AaoVoiceRecOn)

Wert 0/1, gibt an, ob die Spracherkennung derzeit aktiv ist (= PTT gedrückt)

(L:AaoVoiceRecPhrase, String)

Zeichenfolgenvariable, die die aktuell erkannte Phrase enthält. Dies kann beispielsweise bei der Anzeige auf einem FIP nützlich sein, damit Sie die numerische Eingabe mit dem Befehl „back“ korrigieren können.

(L:AaoVoiceRecNum)

Gleitkommazahl, die die endgültige erkannte numerische Eingabe enthält.

(L:AaoVoiceRecNato, String)

Zeichenkette, die die erkannten Buchstaben enthält.

Alle diese Variablen sind AAO-intern; im Simulator kann nicht auf sie zugegriffen werden.

Spracherkennung vs. Diktat

In AAO können Sie auch einen „Diktatmodus“ verwenden. In diesem Modus übergibt die Spracherkennung lediglich eine Zeichenfolgendarstellung Ihrer gesamten Spracheingabe an die Variable „**(L:AaoVoiceRecPhrase, String)**“.

Sie können diese Zeichenfolge dann in einem Skript analysieren oder AAO anweisen, sie in Schaltflächenereignisse zu zerlegen, genau wie bei der oben beschriebenen normalen Spracherkennung.

Um die Diktatfunktion zu nutzen, verwenden Sie diese speziellen AAO-Befehle:

AAO_DICTATION_START: Dieser Befehl startet den Diktatmodus unter Verwendung der aktuell ausgewählten Spracherkennungs-Engine und der aktuellen Ländereinstellung.

AAO_DICTATION_STOP: Dieser Befehl beendet den Diktatmodus, lässt die Diktat-Engine jedoch aktiv.

AAO_DICTATION_RESET: Dieser Befehl beendet die Diktat-Engine und erstellt bei der nächsten Verwendung von „start“ eine neue. Verwenden Sie diesen Befehl, wenn Sie die Sprache ändern.

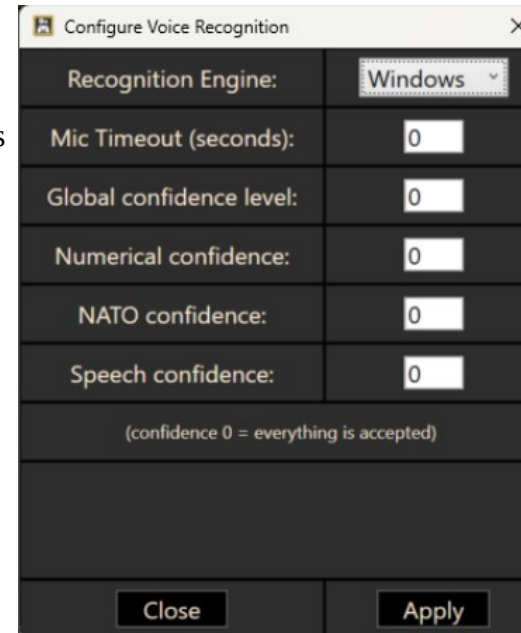
AAO_DICTATION_PARSE: Wenn Sie Sprach-Button-Ereignisse entworfen haben oder Listener-Skripte verwenden, sendet dieser Befehl die diktierete Eingabe zur Verarbeitung an das Spracherkennungsmodul. Er löst alle Button-Ereignisse aus, die er in Ihrer Zeichenfolge finden kann, genau wie bei der normalen Spracherkennung.

Konfiguration der Spracherkennung

Im Dialogfeld „Extras->Spracherkennung konfigurieren“ können Sie die Spracherkennung konfigurieren.

Sie können zwischen der „Windows“- und der „Azure“-Erkennung wählen. Beachten Sie, dass für Azure ein kostenpflichtiges Konto erforderlich ist, das wie im Kapitel „Erweiterte Text-to-Speech-Funktionen“ beschrieben eingerichtet werden muss.

-
- Mikrofon-Timeout: Hiermit wird das Mikrofon nach der angegebenen Zeit automatisch geschlossen (0 = kein Timeout)
-
- Konfidenzstufen (nur Windows-Engine): Hiermit wird die Genauigkeit der Spracherkennung für die verschiedenen Grammatiken festgelegt.



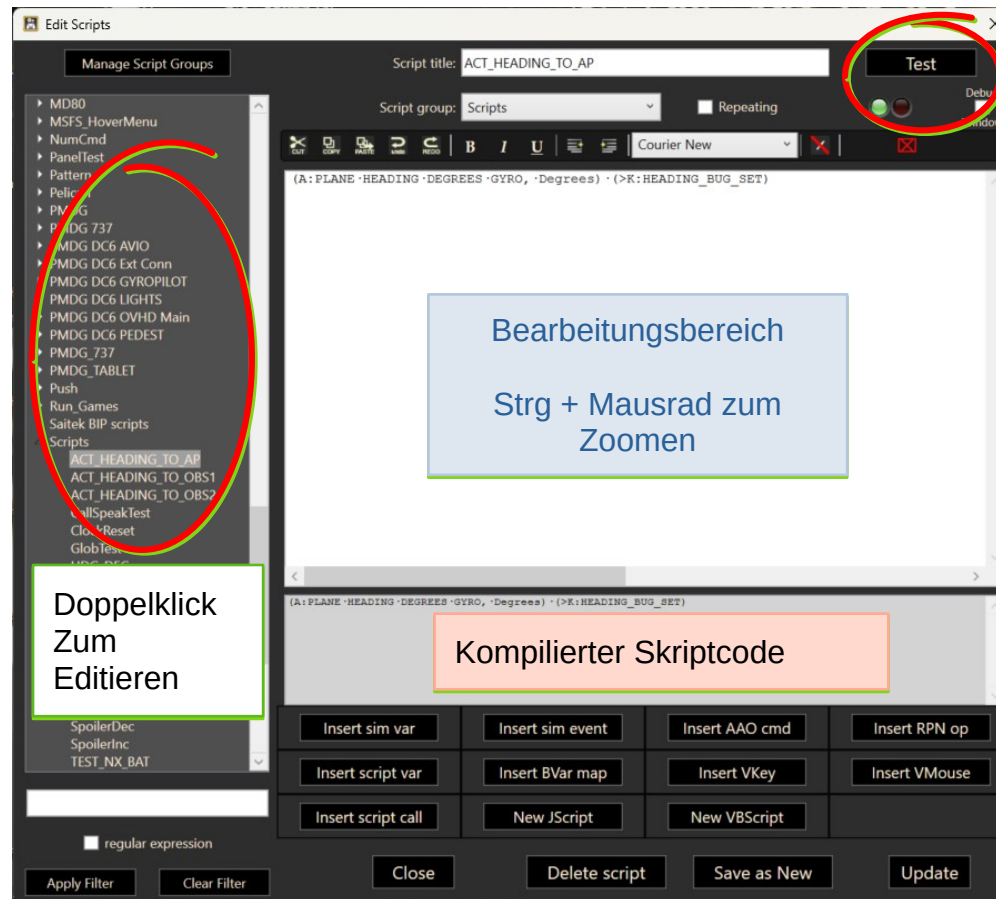
Spracherkennung Desktop/Web FIP

Es gibt eine Standardanzeige „Lorby Voice Recognition Status“, die veranschaulicht, wie diese LVars verwendet werden können. Sie zeigt an, ob AAO gerade zuhört, sowie die erkannte Phrase. Mit dieser Anzeige können Sie die numerische Eingabe mithilfe des Befehls „back“ korrigieren.



5 Skripting

Um komplexe Funktionen auf Schaltflächen nachzubilden, kannst du Skripte erstellen. AxisAndOhs verwendet eine vereinfachte Version der alten FSX-RPN-Notation, aber du kannst auch andere Sprachen verwenden.



Test: Skript an die Verarbeitung senden

Grün: AAO kann das Skript verarbeiten

Rot: Bewegen Sie den Mauszeiger über die LED, um die Fehlerdetails anzuzeigen

Doppelklick
Zum
Editieren

Bearbeitungsbereich
Strg + Mausrad zum
Zoomen

Kompilierter Skriptcode

- Der „Skripttitel“ ist die Bezeichnung, die dieses bestimmte Skript identifiziert. Die Titel müssen für jedes von Ihnen erstellte Skript eindeutig sein.
- „Skriptgruppe“: Sie können die Gruppe ändern, in der das Skript im Dialogfeld zur Ereigniszuweisung aufgeführt wird. Verwenden Sie die Taste „+“, um eigene Gruppennamen hinzuzufügen.
- Ein „Wiederholendes“ Skript führt den Code so lange aus, bis es erneut aufgerufen wird.
- Sie können auf Listen mit Variablen und Ereignissen zugreifen, um diese in das Skript einzufügen.
- Sie können Zeilenkommentare in das Skript einfügen, indem Sie ihnen „//“ voranstellen und sie mit einem Zeilenumbruch beenden.
- Die kleinen Punkte zwischen den Befehlen dienen im RPN-Code lediglich als visuelle Markierung – sie zeigen die Position an, an der sich ein „Leerzeichen“ befindet.

Namenskonvention

Vermeiden Sie die Verwendung der Zeichen „-“ und „:“ in Skript-, Gruppen- und L-Variablenamen!

Das graue Textfeld unten zeigt den eigentlichen Skriptcode an. Bei RPN handelt es sich um eine einzige Textzeile, die keine Zeilenumbrüche enthält.

Die Spezifikation für die MSFS-Variante des RPN-Scriptings ist Teil der SDK-Dokumentation:
[SDK-Dokumentation \(flightsimulator.com\)](https://www.flightsimulator.com/html/SDK-Dokumentation)

Beachten Sie, dass Skripte so einfach wie möglich gehalten werden sollten und dass AAO nicht alle Operatoren unterstützt, sondern nur die gängigsten.

Das Skript einer Schaltfläche zuweisen

The image shows two windows from the Microsoft Flight Simulator SDK configuration tool. The left window, titled 'Add Button Assignment', has the following fields: 'Key Down Event' set to 'ACT_HEADING_TO_AP (ACT_HEADING_TO...', 'Key Up Event' set to 'NONE (None)', 'Custom Label' is empty, 'Virtual key' set to 'None', 'Assigned button/key' set to 'SHIFT CTRL W', 'Assign combo' is 'None', and 'Repeat' is set to 'Med'. The right window, titled 'Select Simulator Event', shows a tree view of events. Under 'A320 PED', the event '0 - A320 PED-ENG MASTER 2 toggle| RPN' is selected and highlighted in blue. A large blue arrow points from the 'Key Down Event' dropdown in the first window to this selected event in the second window.

Durch Drücken der Tastenkombination „Umschalt+Strg+W“ wird nun der aktuelle Kurs als Autopilot-Kurs festgelegt.

Skripte als Ereignisse aufrufen

Skripte können wie Ereignisse aufgerufen werden, indem man „(>K:scriptgroup-scriptname)“ verwendet.

Ein Skript mit Parametern aufrufen

Du kannst Skripten bis zu 99 Parameter übergeben, wenn du sie wie folgt als Ereignisse aufrufst:

„(>K:scriptgroup-scriptname;param1;param2)“.

Innerhalb des aufgerufenen Skripts werden die Zeichenfolgen „param1“, „param2“ usw. durch die Werte ersetzt, die Sie im Aufruf angegeben haben.

Beachten Sie, dass es sich hierbei um eine einfache Textersetzung handelt.

Beispiel: Behandlung einer LVar, die drei Werte annehmen kann: 0, 1 und 2. Dies simuliert einen Dreistufenschalter, der bei jedem Klick auf eine Schaltfläche zwischen 0-1-2-1-0-1-2... wechselt

- Das Skript befindet sich in der Gruppe „Pattern“ und wurde „Lvar_0_1_2_1_0“ genannt
- Der Code des Skripts verwendet zwei Parameter: param1 ist die zu verarbeitende LVar, param2 ist eine lokale LVar, die die Bewegungsrichtung des Schalters speichert (0 = nach oben, 1 = nach unten)

```
(param1) ·s0 ·  
(param2) ·s1 ·  
l0 ·0 ·== ·if{ ·l1 ·0 ·== ·if{ ·1 ·s2 ·} ·els{ ·1 ·s2 ·} ·0 ·s1 ·} ·  
l0 ·1 ·== ·if{ ·l1 ·0 ·== ·if{ ·2 ·s2 ·} ·els{ ·0 ·s2 ·} ·} ·  
l0 ·2 ·== ·if{ ·l1 ·0 ·== ·if{ ·1 ·s2 ·} ·els{ ·1 ·s2 ·} ·1 ·s1 ·} ·  
l2 ·(>param1) ·  
l1 ·(>param2) ·
```

Wenn dieses Skript mit folgendem Befehl aufgerufen wird:

```
1 · (>K:Pattern-Lvar_0_1_2_1_0;L:SWS_FUEL_Switch_Pump_1, Enum;L:SwitchFPDir)
```

wird folgender Code ausgeführt:

```
(L:SWS_FUEL_Switch_Pump_1, Enum) ·s0 ·  
(L:SwitchFPDir) ·s1 ·  
10 ·0 ·== ·if{ ·11 ·0 ·== ·if{ ·1 ·s2 ·} ·els{ ·1 ·s2 ·} ·0 ·s1 ·} ·  
10 ·1 ·== ·if{ ·11 ·0 ·== ·if{ ·2 ·s2 ·} ·els{ ·0 ·s2 ·} ·} ·  
10 ·2 ·== ·if{ ·11 ·0 ·== ·if{ ·1 ·s2 ·} ·els{ ·1 ·s2 ·} ·1 ·s1 ·} ·  
12 · (>L:SWS_FUEL_Switch_Pump_1, Enum) ·  
11 · (>L:SwitchFPDir) ·
```

Verarbeitung des Tasten- oder Achsenwert

Bei Zuweisung zu einer Taste oder Achse kann das Skript den von der Steuerung übermittelten Wert über die Skriptvariable empfangen:

```
(L:Gruppenname-Skriptname)
```

Auf diese Weise können Sie dasselbe Skript für mehrere verschiedene Zuweisungen verwenden. Anstatt beispielsweise zwei Skripte zu benötigen, um die AP-Höhe um 100 oder 1000 zu erhöhen, können Sie einfach das eine verwenden und 100 sowie 1000 als Wert für das Key-Down-Ereignis festlegen:

```
(L:Scripts-MyAltIncScript) (>K:AP_ALT_VAR_INC)
```

Wiederholen eines >K:-Ereignisses

Durch Hinzufügen des Pipe-Zeichens „|“ können Sie ein Ereignis oder Skript mehrfach (n-mal) wiederholen:

```
x (>K:Ereignis- oder Skriptname|n)
```

>K:n:-Ereignisse mit mehreren Eingabewerten

In MSFS können K-Ereignisse mit mehr als einem Eingabewert aufgerufen werden. Derzeit sind bis zu 5 Parameter möglich. Einzelheiten finden Sie in der MSFS-SDK-Dokumentation.

Die Syntax hängt von der Anzahl der Parameter ab:

```
val1 val0 (>K:2:eventname)
```

```
val2 val1 val0 (>K:3:eventname)
```

```
val3 val2 val1 val0 (>K:4:eventname)
```

```
val4 val3 val2 val1 val0 (>K:5:eventname)
```

Beispiel:

```
0 1500 (>K:2:AP_ALT_VAR_SET_ENGLISH)
```

Dies setzt 1500 Fuß in Slot 0 des Autopiloten.

Das Ergebnis eines Skripts auslesen

Auf Skripte, die einen Wert (eine Zahl oder eine Zeichenkette) zurückgeben, kann von anderen Skripten aus wie folgt zugegriffen werden:

```
(S:Gruppenname-Skriptname)
```

```
(S:Gruppenname-Skriptname, Zeichenkette)
```

Beispiel:

Das Berechnungsskript befindet sich in der Gruppe „testgroup“, heißt „add“ und hat den Code „8 5 +“.

Dann kann ein anderes Skript das Ergebnis dieser Berechnung aufrufen:

```
(S:testgroup-add) 10 + (>L:mytestresult)
```

Durch Ausführen dieses Skripts wird „23“ in die Lvar geschrieben.

Lokale Variablen (L: vars):

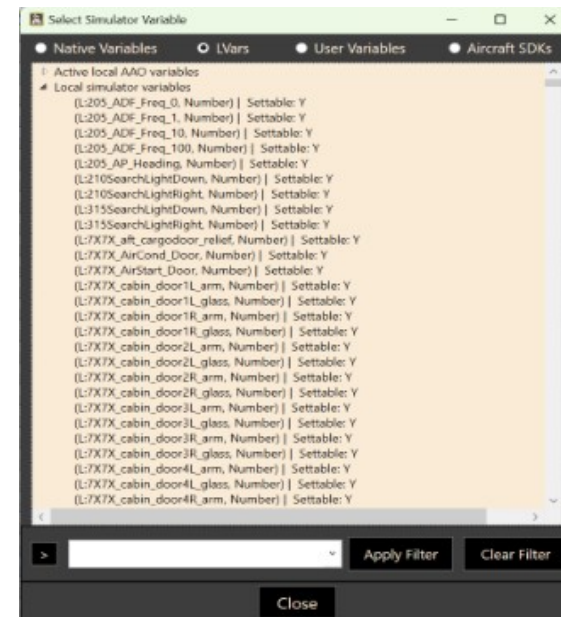
Mit AAO-Skripten können Sie auch lokale Variablen lesen und schreiben

- Wenn die Variable nur lokal für AAO gelten soll, geben Sie keine Einheit an: (L:AaoLocalVar)
- String-Variablen sind ebenfalls lokal für AAO: (L:AaoStringVar, String)
- Wenn es sich bei der Variablen hingegen um eine LVAR aus dem Simulator handelt, muss die Einheit angegeben werden (außer String!): (L:SimLocalVar, Number)

Eine Liste aller LVARs aus dem Simulator abrufen

Verwenden Sie die Menüoption „Scripting → Read LVARs from sim“, um eine Liste aller LVARs zu importieren, die derzeit im Simulator aktiv sind. Das Ergebnis ist im Dialog „RPN Scripts Editor“ über die Schaltfläche „Insert variable“ verfügbar: Alle erfassten LVARs werden in der Gruppe „Local simulator Variables“ gesammelt.

LVARs werden in dieser Liste nur angezeigt, wenn sie mindestens einmal im Simulator verwendet wurden.



HTML-Variablen (H: vars):

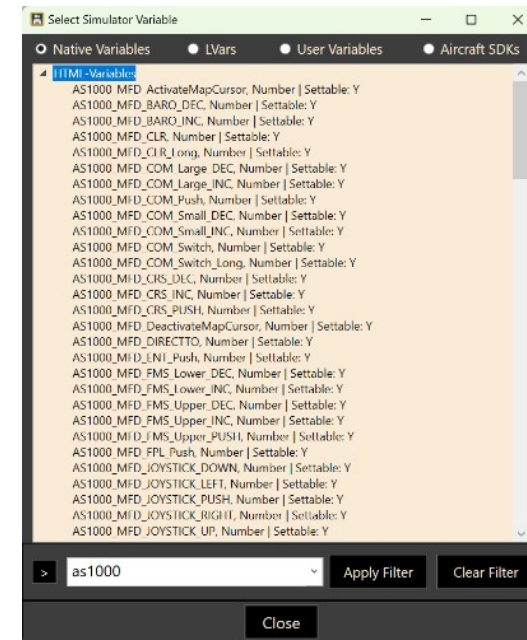
HTML-Variablen sind in der Regel Teil der Animationsdefinitionen eines Flugzeugs. Sie können verwendet werden, um Bedienelemente auf den HTML-Instrumenten zu steuern, auf die über normale Simulatorereignisse nicht zugegriffen werden kann.

Abrufen einer Liste aller H: vars aus dem Simulator

Verwenden Sie die Menüoption „Scripting → Read HVARs from sim“, um eine Liste aller H: vars zu importieren, die in den verschiedenen Definitionsdateien des Simulators zu finden sind.

Dies funktioniert nur, wenn AAO nicht mit dem Simulator verbunden ist!

Das Ergebnis ist im Dialog „RPN-Skripte verwalten“ über die Schaltfläche „Variable einfügen“ verfügbar: Alle erfassten Variablen werden in der Gruppe „HTML-Variablen“ gesammelt.



Einige dieser HTML-Ereignisse enthalten Tags, die von „#“-Zeichen umschlossen sind. Diese Zeichenfolgen müssen durch tatsächliche Werte ersetzt werden, damit das Ereignis funktioniert. Eine Liste der wichtigsten Tag-Werte finden Sie in der Liste „HTML-Tags“; Sie können einfach darauf doppelklicken, um das Tag zu ersetzen. IDs sind standardmäßig meist numerische Werte.

B Eingabeereignisse (B: vars):

In Microsoft Flight Simulator gibt es eine weitere Möglichkeit, Eingaben zu verarbeiten: die „BVars“. BVars haben keine festen Namen, sondern werden von den Entwicklern an die jeweiligen Anforderungen angepasst. Um diese Eingabeereignisse zu ermitteln, musst du den Entwicklermodus von MSFS nutzen. BVars können sowohl beschrieben als auch gelesen werden.

Verwenden Sie in RPN-Skripten

(B:Variablenname, Einheit) zum Lesen und n (>B:Variablenname_Aktion) zum Schreiben (keine Einheit beim Schreiben!)

```
1 (B:SomeSwitchCover, Bool) - (>B:SomeSwitchCover_Set)
1 (>B:SomeSwitchCover_Toggle)
1 (>B:SomeRotaryKnob_INC)
1 (>B:SomeRotaryKnob_DEC)
```

Beim Schreiben in BVars können Sie bis zu 5 Parameter verwenden. Beispiel:

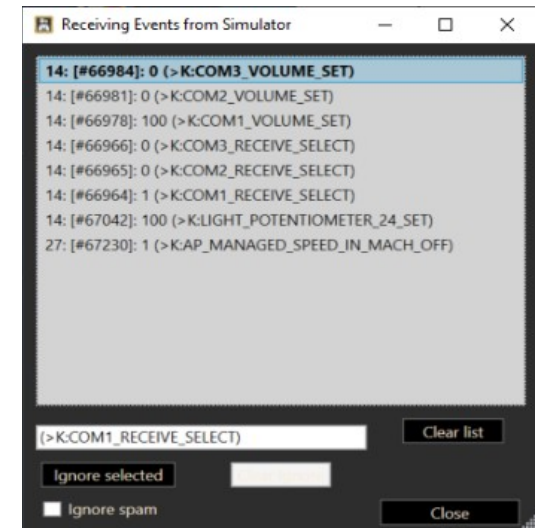
```
0.5·1·(>B:AS1000_RANGE_MFD_Set)
```

Beispielskript für die Cessna C208, das die rote Abdeckung des externen Netzschalters zweimal nach oben und unten klappt, wobei verschiedene Varianten derselben BVar verwendet werden:

```
0·(>B:ELECTRICAL_ExternalPower_1_Cover_Set)·1·(>B:ELECTRICAL_ExternalPower_1_Cover_Set)·(SPLIT:1000)·
0·(>B:ELECTRICAL_ExternalPower_1_Cover_Set)·(SPLIT:1000)·0·(>B:ELECTRICAL_ExternalPower_1_Cover_Open)
·(SPLIT:1000)·0·(>B:ELECTRICAL_ExternalPower_1_Cover_Close)
```

Verfolgen der Standard-Simulatorereignisse

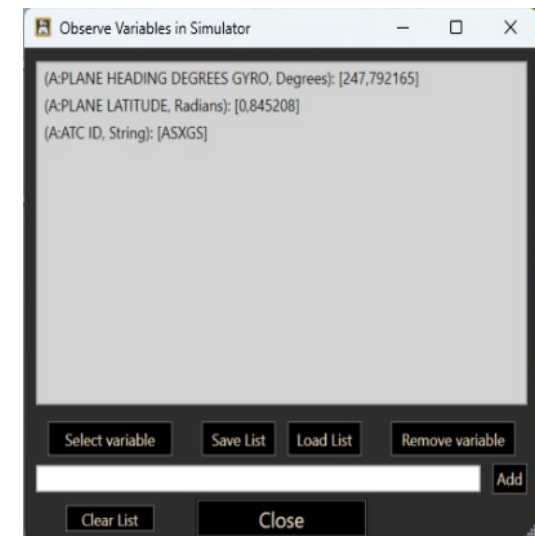
Öffnen Sie den Ereignis-Beobachter über den Menüpunkt „Scripting → Watch simulator events“. In diesem Dialog werden Simulatorereignisse angezeigt, sobald sie ausgelöst werden – beispielsweise wenn Sie im VC auf eine Schaltfläche klicken. Beachten Sie jedoch, dass nicht alle Schaltflächen mit Simulatorereignissen verknüpft sind. Je komplexer ein Flugzeug ist, desto höher ist die Wahrscheinlichkeit, dass der Entwickler andere Mittel zur Schaltflächenbetätigung gewählt hat. In vielen Fällen sendet die Flugzeuglogik selbst Ereignisse, manchmal sogar sehr viele. Verwenden Sie das Kontrollkästchen „Spam ignorieren“, um alle Ereignisse automatisch zu ignorieren, die mehr als 40 Mal empfangen wurden.



Simulatorvariablen verfolgen

Verwenden Sie die Menüoption „Scripting → Simulatorvariablen beobachten“, um den Variablen-Watcher zu öffnen.

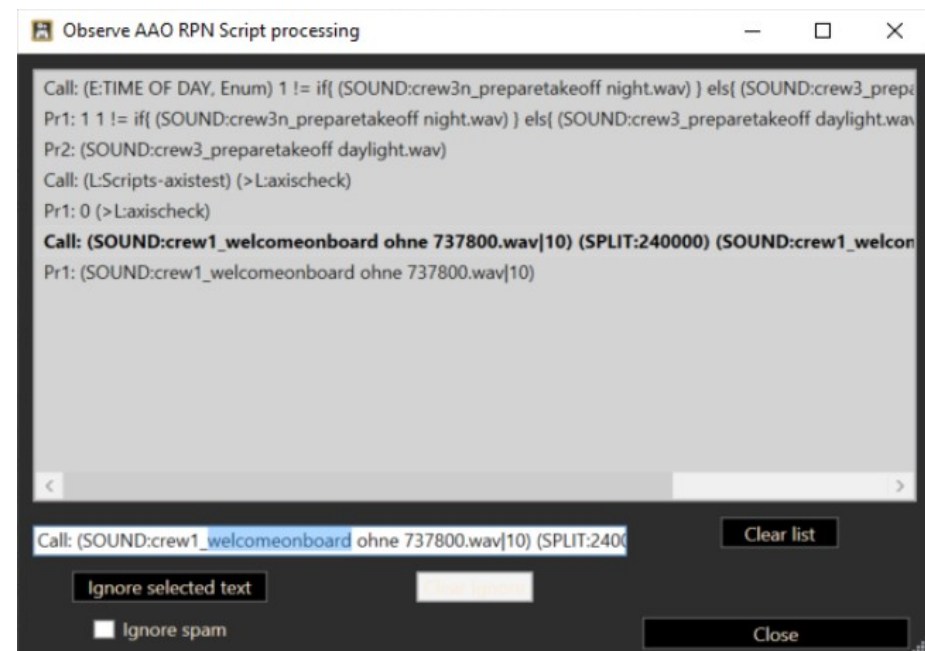
1. Sie können Simulatorvariablen aus der üblichen Baumliste über „Variable auswählen“ hinzufügen oder deren Namen in das Textfeld eingeben.
2. Klicken Sie anschließend auf „Hinzufügen“, um die Variable zur Liste hinzuzufügen.
3. Die Liste kann in einer Textdatei gespeichert bzw. aus einer Textdatei geladen werden.



Ausführung von Skripten verfolgen

Öffnen Sie die Skript-Handler-Konsole über den Menüpunkt „Skripting → AAO-Skriptverarbeitung beobachten“.

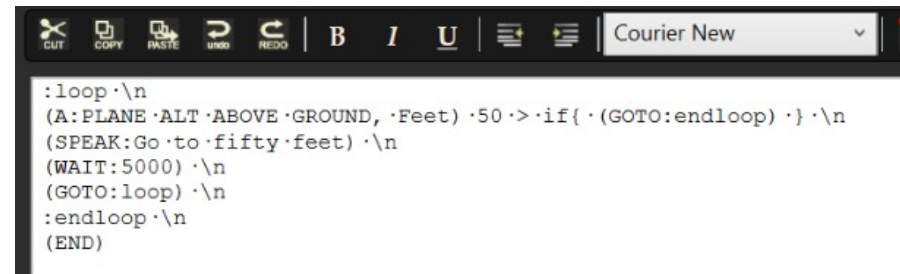
- Aufruf: Der Skriptcode, der in den Skript-Handler gelangt
- Pr1/Pr2 sind Verarbeitungsschritte
- „Spam ignorieren“ ignoriert ein Skript automatisch, wenn derselbe Code mehr als 20 Mal erkannt wird
- Klicken Sie auf ein Skript, um es in das Ignorierfeld zu verschieben
- Mit „Ausgewählten Text ignorieren“ können Sie AAO anweisen, alle Skripte zu ignorieren, die den im Feld stehenden Text enthalten, oder nur den Teil, den Sie im Feld mit der Maus markieren



Skripte mit mehreren Zeilen:

Normalerweise besteht ein RPN-Skript nur aus einer einzigen Codezeile. In AAO können Sie Skripte mit mehreren Zeilen erstellen, sodass Sie Befehle wie GOTO verwenden können. Mehrzeilige Skripte werden intern in Objekte vom Typ CONVERSATION umgewandelt (siehe Kapitel über „RPN-Skriptdateien“).

Um ein Skript in mehrere Zeilen aufzuteilen, fügen Sie im RPN-Editor am Ende jeder Zeile „\n“ ein:



```
:loop·\n
(A:PLANE·ALT·ABOVE·GROUND,·Feet)·50·>·if{·(GOTO:endloop)·}·\n
(SPEAK:Go·to·fifty·feet)·\n
(WAIT:5000)·\n
(GOTO:loop)·\n
:endloop·\n
(END)
```

Skripte exportieren und importieren:

Diese Funktion dient dem Austausch von Skripten mit anderen Benutzern

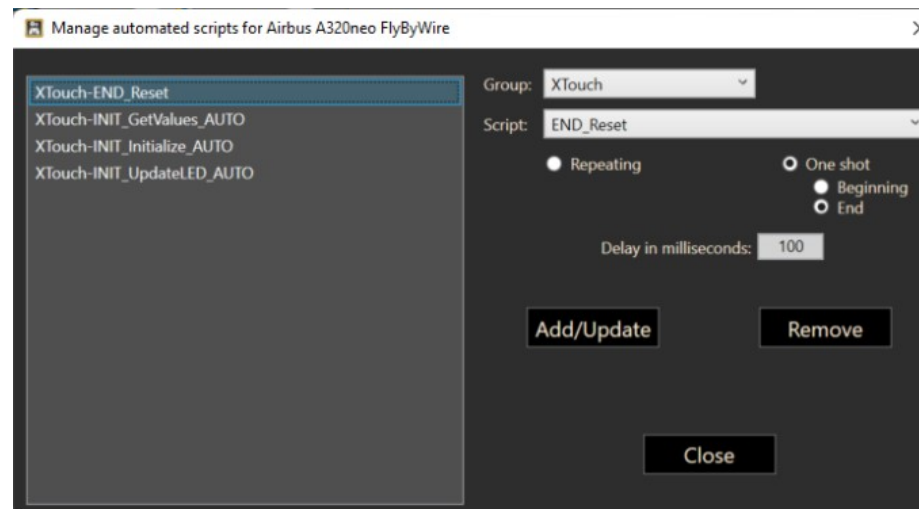
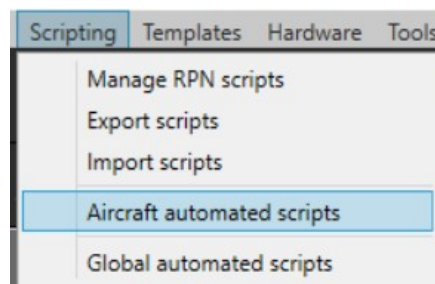
- Über den Dialog „Scripting->Skripte exportieren“ können Sie Ihre Skripte in eine einfache XML-Datei schreiben.
- Über „Scripting->Skripte importieren“ können Sie die Skripte aus einer XML-Datei in Ihre lokale AAO-Datenbank einlesen. Der Skriptimport ist nur möglich, wenn AAO nicht mit dem Simulator verbunden ist (= grüne LED leuchtet nicht).

Automatisierte Skripte:

Sie können Skripte so einstellen, dass sie automatisch ausgeführt werden, ohne dass Sie eine Taste oder einen Regler betätigen müssen. Diese Einstellung kann global vorgenommen werden, sodass das Skript unabhängig vom aktuellen Flugzeug ständig läuft. Oder Sie können automatisierte Skripte so einstellen, dass sie nur bei einem bestimmten Flugzeug ausgeführt werden.

- „Wiederholende“ Skripte werden jedes Mal ausgeführt, wenn die „Verzögerung“ abgelaufen ist.
- Ein „One-Shot“-Skript wird nach der „Verzögerung“ nur einmal ausgeführt. Sie können wählen, ob es beim Laden des Flugzeugs („Beginn“) oder beim Beenden der Simulatorsitzung („Ende“) ausgeführt werden soll.

Die „Verzögerung“ kann mit dem Mausrad geändert werden.



RPN-Makros

In AAO können Sie eine vereinfachte Version der XML-Gauge-Makros verwenden. Makros dienen der einfachen Textersetzung:

→ Makrodefinition

```
<Macro Name="strobelightmacro">A:LIGHT STROBE, Bool</Macro>
```

→ Verwendung des Makros in Ihren RPN-Skripten:

```
(@strobelightmacro) 1 == if{ 1 (>K:STROBES_TOGGLE) }
```

→ Ergebnis zur Laufzeit:

```
(A:LIGHT STROBE, Bool) 1 == if{ 1 (>K:STROBES_TOGGLE) }
```

Bitte beachten Sie, dass Makrodefinitionen in eigenen Skripten gespeichert werden müssen. Sie sind statisch und können nicht mit RPN-Code gemischt werden, der dynamisch geparkt werden müsste.

Sie können auch LVars als Makros verwenden, AAO ersetzt dabei „@variablename“ durch den tatsächlichen Wert:

```
2 · (>L:engnum)  
(A:ENG·COMBUSTION:@engnum, ·Bool)  
=> (A:ENG·COMBUSTION:2, ·Bool)
```

RPN-Operatoren in AAO

Operator	Operation	Argumente	Beispiel	Ergebnis
Gängige Operatoren				
+	Addition	2	3 6 +	8
-	Subtraktion. Wenn der Stapel A B – enthält, dann lautet die Berechnung A - B.	2	(L:Value) 90 -	Der lokale Wert minus 90
/	Division. Wenn der Stapel A B / enthält, dann lautet die Berechnung A / B.	2	5 2 /	2.5
*	Multiplication	2	Pi 2 *	2 Pi
% mod	Modulo-Operation	2	5 3 mod	2
++	Inkrement	1	4++	5
--	Decrement	1	4 --	3
/-/ neg	negiert eine Zahl	1	4 /-/ neg	-
Vergleichsoperatoren				
==	Wahr wenn gleich	2	(L:Value) 0 == if{ A }	Operation A wird ausgeführt, wenn der Wert 0 ist.
!=	Wahr wenn nicht gleich	2	(L:Value) 0 != if{ A }	Operation A wird ausgeführt, wenn der Wert nicht 0 ist.
>	Wahr wenn größer	2	(L:Value1) (L:Value2) > if{ A } els{ B }	Ist Wert1 größer als Wert2, wird Operation A ausgeführt, andernfalls wird Operation B ausgeführt.
<	Wahr wenn kleiner	2	(L:Value1) (L:Value2) < if{ A } els{ B }	Ist Wert1 kleiner als Wert2, wird Operation A ausgeführt, andernfalls wird Operation B ausgeführt.
>=	Wahr wenn größer oder gleich	2	(L:Value1) (L:Value2) >= if{ A } els{ B }	Wenn Wert1 größer oder gleich Wert2 ist, wird Operation A ausgeführt, andernfalls wird Operation B ausgeführt.
<=	Wahr wenn kleiner oder gleich	2	(L:Value1) (L:Value2) <= if{ A } els{ B }	Wenn Wert1 kleiner oder gleich Wert2 ist, wird Operation A ausgeführt, andernfalls wird Operation B ausgeführt.

Operator	Operation	Argumente	Beispiel	Ergebnis
?	Der dritte Operand bestimmt, ob der erste (True) oder der zweite (False) ausgewählt wird.	3	A B True ?	Das ergibt A.
Bit Operatoren				
&	bitweises AND	2	5 3 &	1
	bitweises OR	2	5 3	7
^	bitweises XOR	2	5 3 ^	6
~	bitweises NOT	1	5 ~	-6
>>	um die Anzahl der Bits nach rechts verschieben	2	5 3 >>	0
<<	um die Anzahl der Bits nach links verschieben	2	5 3 <<	40
bytof	Bitweise Umwandlung von „double“ in „float“	2	1116892707587883008.000·4·bytof	350
bytoi	Bitweise Umwandlung von „double“ in 32bit int	3	1116892707587883008.000·0·bytoi	0
setbit	ein bestimmtes Bit in einer Zahl auf 1 setzen	3	0x0000 0 1 setbit	2
clrbit	ein bestimmtes Bit in einer Zahl auf 1 setzen	3	0x0011 0 0 clrbit	16
Bitoperatoren für Byte-Arrays = Hexadezimalzeichenfolgen („22AACFFDD55“)				
&s	bitweises AND	2	'0A00000001' '0000000001' &s	'0000000001'
s	bitweises OR	2	'0A00000001' '0000000001' s	'0A00000001'
^s	bitweises XOR	2	'0A00000001' '0000000001' ^s	'0A00000000'
~s	bitweises NOT	1	'0A00000001' ~s	'F5FFFFFFFE'
>s	um die Anzahl der Bits nach rechts verschieben	2	'0A00000001' 5 >s	'0050000000'
<s	um die Anzahl der Bits nach links verschieben	2	'0A00000001' 5 <s	'4000000020'
setbit	ein bestimmtes Bit in einer Zahl auf 1 setzen	3	'010000000001'·4·5·setbits	'012000000001'
clrbit	ein bestimmtes Bit in einer Zahl auf 1 setzen	3	'010000000001'·0·0·clrbits	'010000000000'

Operator	Operation	Argumente	Beispiel	Ergebnis
Logische Operatoren				
!, not	not	1	(L:Local) ! (>L:Local)	Schaltet die Variable „Local“ um
&&, and	and	2	(L:Local) 0xFF00 && (>L:Local)	Die Variable „Local“ wird mit dem Hexadezimalwert 0xFF00 UND verknüpft
, or	or	2	(L:Local) 07777 OR (>L:Local)	Die Variable „Local“ wird mit der Oktalzahl 7777 logisch ODER verknüpft.
Numerische Operatoren				
abs	Absolutwert	1	-5 abs	5
int flr	Berechnet die nächste ganze Zahl, die kleiner ist als die Eingabezahl	1	5.98 flr	5
rng irng	Bereich; gibt „True“ zurück, wenn der dritte Operand zwischen den Werten eins und zwei liegt. „irng“ schließt die Grenzen mit ein	3	4 7 6 rng 4 7 4 rng 4 7 4 irng	True False True
cos	Cosinus (Eingabe in Bogenmaß)	1	pi cos	-1
lg	Logarithmus zur Basis 10	1	10 lg	1
min	Minimum	2	5 2 min	2
sin	Sinus (Eingabe in Bogenmaß)	1	pi sin	0
acos	Arcuscosinus (Ausgabe in Bogenmaß)	1	pi acos	
ctg	Cotangens (Eingabe in Bogenmaß)	1	pi ctg	
ln	Natürlicher Logarithmus	1	2.718282 ln	1
sqr	Quadrat	1	5 sqr	25
asin	Arcussinus	1	pi asin	
eps	Relative Genauigkeit bei Gleitkommazahlen	1	1 eps	2 ⁽⁻⁵²⁾
log	Logarithmus von Operand eins zur Basis von Operand zwei.	2	8 2 log	3

Operator	Operation	Argumente	Beispiel	Ergebnis
pi	Pi = 3,14159; legt Pi auf den Stapel	0	pi	3.14159
sqrt	Quadratwurzel	1	25 sqrt	5
atg2	Arcustangens mit zwei Eingaben (Eingabe in Bogenmaß)	2		
exp	Exponent; e hoch Operand	1	1 exp	2.718282
max	Maximum	2	5 2 max	5
pow	Potenz von; der erste Wert hoch zwei	2	2 5 pow	32
tg	Tangens (Eingabe in Bogenmaß)	1	pi tg	0
atg	Arcustangens mit einem Eingabewert	1	pi atg	
Spezielle Operatoren				
rnd	Erzeugt einen zufälligen ganzzahligen Wert	1	10 rnd	Ein Zufallswert zwischen 0 und 9
div	Teilt ganze Zahlen; das Ergebnis ist immer eine ganze Zahl	2	5 3 div	1
ceil	Berechnet die nächste ganze Zahl, die größer ist als der Ausgangswert	1	4.3 ceil	5
near	Berechnet die nächste ganze Zahl, wobei auf 0,5 aufgerundet wird	1	4.5 near	5
dnor d360 rdeg	Normalisiert einen in Grad angegebenen Winkel. Das Ergebnis ist ein Wert zwischen 0 und 360.	1	-15 dnor	345
rddg	Rechnet Bogenmaß in Grad um	1	pi rddg	180
dgrd	Rechnet Grad in Bogenmaß um	1	180 dgrd	pi
rnor	Normalisiert einen in Bogenmaß ausgedrückten Winkel; das Ergebnis dieser Operation liegt zwischen 0 und 2π	1	5 rnor	1.8584

Operator	Operation	Argumente	Beispiel	Ergebnis
ddiff	Berechnet die Winkelabweichung zwischen zwei Kursen (in Grad)	2	270 45 ddiff	-135
lldist	Entfernung zwischen zwei Punkten (Breitengrad/Längengrad)	4	40.51·9.21·42.43·10.44·lldist	127.85 Nautische Meilen
llbrg	Peilung von einem Längen- und Breitengrad zu einem anderen	4	40.51·9.21·42.43·10.44·llbrg	25.64 Grad
dec2b16	Konvertiert Dezimalzahlen in die BCD16-Kodierung, die in Ereignissen zur Frequenzeinstellung wie (>K:COM_RADIO_SET) verwendet wird	1	13025 dec2b16	77861
ar429	ARINC429-Wert mit 4 Bytes konvertieren	1	14023863173 ar429	454.3399
unixts unixtsms	Konvertieren einer „Epoch“ (Anzahl der Sekunden seit dem 01.01.1970) mithilfe einer Formatzeichenfolge Die Formatzeichenfolge kann ein beliebiges C#-Standard-Datumsformat sein oder die Zeichen ‚d‘, ‚M‘, ‚y‘, ‚h‘, ‚m‘, ‚s‘ geben das angegebene Zeitstempелеlement als Zahl zurück „unixtsms“ kann zusätzlich ‚fff‘ verwenden	2	1685433887 'dd.MM.yyyy HH:mm:ss' unixts	'30.05.2023 08:04:47'
time	Aktueller Systemzeitstempel in Millisekunden	0	time 'HH:mm:ss' unixtsms	'16:11:23.564
lvar	Den Wert einer benannten LVar zurückgeben	1	'(L:mylvar)' lvar '(L:mystrvar, String)' lvar	0 'test'
if{ }	If-Anweisung: Beachten Sie, dass zwischen dem „if“ und dem „{“ kein Leerzeichen steht	1	(L:Value) 0 == if{ A }	Operation A wird ausgeführt, wenn der Wert 0 ist.
els{ }	else-Anweisung: Beachten Sie, dass zwischen „else“ und „{“ kein Leerzeichen steht.	1	(L:Value1) (L:Value2) <= if{ A } els{ B }	Wenn Wert1 kleiner oder gleich Wert2 ist, wird Operation A ausgeführt, andernfalls B
rndsel{...}	Zufällige Auswahl – bitte beachten Sie, dass zwischen „rndsel“ und „{“ kein Leerzeichen steht. Innerhalb der geschweiften Klammern {} dürfen sich nur (>K-Ereignisse oder AAO-Befehle	n	rndsel{ (>K:PAUSE_TOGGLE) (SOUND:crew3_preparetakeoff daylight.wav) (SPEAK:Third alternative) (SPEAK:Fourth	Je nach Ergebnis der Zufallsauswahl wird die Simulation angehalten, die Sounddatei abgespielt oder das Sprachsystem gibt einen der beiden Sätze wieder

	befinden. Die App wählt eines davon nach dem Zufallsprinzip aus.		alternative) }	
--	--	--	----------------	--

Operator	Operation	Argumente	Beispiel	Ergebnis
quit	Die Anweisung „quit“ sorgt dafür, dass die Auswertung des Ausdrucks vollständig beendet wird, und vermeidet die Verwendung verschachtelter if-Anweisungen.	0	pi quit (L:Value1) (L:Value2) <= if{ A } els{ B }	pi. Der Rest des Skripts wird ignoriert.
g0...gn	Zum Label springen. Die Ausführung springt zum angegebenen Label. Labels werden durch Eingabe eines Doppelpunkts gefolgt von der Labelnummer festgelegt.	0	g4	Ausführung bei :4 fortsetzen
case	Case-Anweisung [Liste der Rückgabewerte] [Anzahl der Werte] [eval-Wert] case		50 40 30 20 10 5 (A:var, unit) case	Die „5“ gibt an, dass es fünf Fallwerte gibt, die je nach dem Ergebnis der Auswertung von (A:var) ausgewählt werden. Ist das Ergebnis größer oder gleich 0, aber kleiner als 1, lautet das Ergebnis 10. Ist das Ergebnis größer oder gleich 1, aber kleiner als 2, lautet das Ergebnis 20 und so weiter.
nonlin	Nichtlineare Auswahl [Liste der zu vergleichenden Werte] [Anzahl der Werte] [Bewertungswert] nonlin		10 15 30 42 100 5 (A:var, unit) nonlin	Die „5“ gibt an, dass es fünf Werte gibt, die mit (A:var) verglichen werden. Der Operator gibt den Index des Werts zurück, der größer oder gleich (A:var) ist, in diesem Fall 0–4
nonlin{ }map	Nichtlineare Interpolation [eval value] nonlin{ [x/y-Paare] }map	N + 1	600 nonlin{ 0 0 100 3 500 103 800 252 1000 257 }map	Ergebnis: 152,6 Die x,y-Zuordnung lautet (0,0), (100,3), (500,103), (800,252), (1000,257) Der Operator sucht nach dem x-Wert, der größer oder gleich dem eval-Wert ist, und interpoliert die y-Position zwischen diesem und dem vorhergehenden Wert.
seq	Sequenzoperator. Gibt den Wert zurück, der nach dem Vergleichswert an der Reihe ist [Liste der zu vergleichenden Werte] [Eval-Wert] [Anzahl der Werte] seq			

Operator	Operation	Argumente	Beispiel	Ergebnis
iseq	Operator für indizierte Sequenzen. Gibt den Wert zurück, der nach dem Vergleichswert an einem bestimmten Index an der Reihe ist. AAO-Befehl: (VARSEQ)	N + 3	0 1 2 1 (L:var) (L:idx) 4 iseq (>L:var) (>L:idx) or using the shortcut: (VARSEQ:0~1~2~1 L:var)	Advances L:var through the sequence 0 - 1 - 2 - 1 - 0 etc. Current index is stored in L:idx, this must be a unique variable name (the shortcut does this automatically)
n iterate{ rpncode }next	rpncode n-mal wiederholen	1	3 iterate{ 1 (>K:ELEV_TRIM_UP) }next	Die Höhenverstellung wird dreimal durchgeführt
procexist	Prüfen, ob ein Prozess läuft	1	'flightsimulator' procexist	1, wenn MSFS läuft, andernfalls 0
winfocus	Den Titel des aktuell fokussierten Fensters abrufen	0	winfocus (>L:currentWindow, String)	
filexist	Prüfen, ob eine Datei vorhanden ist (absolut oder relativ zum AAO-Dokumentenpfad)	1	'textfile.txt' filexist	1 wenn die Datei existiert
direxist	Prüfen, ob ein Ordner vorhanden ist (absolut oder relativ zum Pfad der AAO-Dokumente)	1	'mysubdir' direxist	1 wenn die Verzeichnis existiert
mousex	Horizontale Mausposition abrufen	0	mousex	1288
mousey	Vertikale Mausposition abrufen	0	mousey	632
mousecmp	Prüfen, ob sich der Mauszeiger in einem bestimmten rechteckigen Bereich befindet	4	1000 100 500 50 mousecmp	1, wenn sich der Mauszeiger im Bereich 1000,500 – 1100,550 befindet, andernfalls 0
nlog	Schreibe den numerischen Stapel in die Skript-Protokolldatei	0	5 7 9 nlog	Schreibt „9.0000 7.0000 5.0000“ in die Protokolldatei unter „Dokumente\LorbyAxisAndOhsFiles\Scripts“
slog	Schreibe den String-Stack in die Skript-Protokolldatei	0	'text1' 'text2' slog	Schreibt „text2 text1“ in die Protokolldatei unter „Dokumente\LorbyAxisAndOhsFiles\Scripts“
Zeichenkettenoperatoren				
'...'	Bezeichnet ein String-Literal im Skript (char39, ALT&0-3-9) Wenn innerhalb des Strings ein Apostroph verwendet wird, muss dieser durch „&apos;“ ersetzt werden		'This is a test' (>L:MyTestVar, String)	

Operator	Operation	Argumente	Beispiel	Ergebnis
lc	Konvertiert Zeichenkette in Kleinschreibung	1	'ABcd10' lc	abcd10
uc, cap	Konvertiert Zeichenkette in Großschreibung	1	'Abcd10' uc	ABCD10
chr	Konvertiert eine Ziffer in ein Zeichen	1	65 chr	A
ord	Konvertiert ein Zeichen in eine Integerzahl	1	'A' ord	65
scat	Verbindet Zeichenketten	2	'abc' 'red' scat	abcred
scmp	Vergleicht Zeichenfolgen unter Berücksichtigung der Groß-/Kleinschreibung	2	'left' 'Left' scmp if{ 'yes' } els{ 'no' }	no
slen	Legt die Länge einer Zeichenkette auf den Stapel	1	'right' slen	5
stod	Wandelt eine numerische Zeichenfolge in eine Zahl um	1	'123' stod (>K:...)	123 (>K:...)
scmi	Vergleicht Zeichenfolgen, wobei die Groß-/Kleinschreibung ignoriert wird	2	'left' 'Left' scmi if{ 'yes' }	yes
sstr	Sucht eine Teilzeichenfolge	2	'cd' 'abcde' sstr	2
scon	Wahr, wenn die zweite Zeichenkette die erste enthält	2	'320' 'Airbus A320' scon	1
substr	Extrahiert eine Teilzeichenfolge an der Position a mit der Länge b	3	'abcdef' 2 3 substr	'cde'
ssub	Extrahiert eine Teilzeichenfolge	2	'ab' 'abcde' ssub	'cde'
symb	Extrahiert ein einzelnes Zeichen	2	'abc' 1 symb	'b'
srep	Teile einer Zeichenfolge ersetzen	3	'abcde' 'bc' 'xy' srep	'axyde'
schr	Position einer Zeichenfolge innerhalb einer Zeichenfolge ermitteln	2	'12345' '3' schr	2
ssplit	Eine Zeichenkette anhand einer Teilzeichenkette aufteilen und einen bestimmten Wert an der angegebenen Position zurückgeben. Bei einem Index von -1 werden alle Elemente in den Stapel von Zeichenketten zerlegt.	3	'ABC DEF GHI' '' 1 ssplit 'ABC DEF GHI' '' -1 ssplit	'DEF' stack: 'ABC' 'DEF' 'GHI'
asplit	Eine Zeichenkette anhand einer Teilzeichenkette aufteilen und die Ergebnisse in ein String-Array schreiben	3	'ABC DEF GHI' '' 'arrayname' asplit	(STRARR:arrayname) enthält die Werte ABC, DEF und GHI
shex	Eine Zeichenkette in ein ASCII-Hex-Array umwandeln	1	'ABCD' shex	'41424344'

Operator	Operation	Argumente	Beispiel	Ergebnis
nhex	Eine Zahl in ein Hex-Array umwandeln		12500 nhex	'0102050000'
%...%!...!	ahlenformatierung: d = Ganzzahl, f = Gleitkommazahl. Setze vor „d“ eine 0 oder ein Leerzeichen, um führende Nullen oder Leerzeichen einzufügen. %...% kann RPN-Code und Variablen enthalten		%2%!02d! %12.34554%!4.2f!	02 12.35
Stapeloperatoren				
c	Leert den numerischen Stapel	0	stack: 1 2 3 c	stack:
d	Dupliziert den Wert, der ganz oben steht	1	stack: 5 d	Stack:5 5
p	Holt den obersten Wert ab und verwirft ihn	1	stack: 1 2 3 p	Stack:1 2
r	Vertauscht den ersten und den zweiten Wert	2	stack: 1 2 3 r	Stack:1 3 2
cs	Leert den String-Stack	0	stack: 'st' 'ab' 'xy' cs	stack:
ds	Dupliziert den Wert, der ganz oben steht	1	stack: 'st' ds	stack: 'st' 'st'
ps	Holt den obersten Wert ab und verwirft ihn	1	stack: 'st' 'ab' 'xy' ps	stack: 'st' 'ab'
rs	Vertauscht den ersten und den zweiten Wert	2	stack: 'st' 'ab' 'xy' rs	stack: 'st' 'xy' 'ab'
Numerische Register				
s0 ... s99 gs0 ... gs99	Speichert den Wert auf dem numerischen Stack in einem lokalen oder globalen Register; der Wert verbleibt auf dem Stack.	1	stack: 1 2 3 s0	stack: 1 2 3 s0: 3
l0 ... l99 gl0 ... gl99	Lädt einen Wert aus einem Register an den Anfang des numerischen Stacks	1	stack: 1 2 3 s0 l0	stack: 1 2 3 3
sp0 ... sp99 gsp0 ... gsp99	Speichert den obersten Wert und entfernt ihn vom Zahlenstapel	1	stack: 1 2 3 sp0	stack: 1 2 s0: 3
Zeichenkettenregister				
ss0 ... ss99 gss0 ... gss99	Speichert den Wert auf dem String-Stack in einem lokalen oder globalen Register; der Wert verbleibt auf dem Stack	1	stack: 'A' 'B' 'Test' ss0	stack: 'A' 'B' 'Test' ss0: 'Test'

Operator	Operation	Argumente	Beispiel	Ergebnis
sI0 ... sI99 gsl0 ... gsl99	Lädt einen Wert aus einem Register an den Anfang des Zeichenfolgenstapels	1	stack: 'A' 'B' 'Test' ss0 sI0	stack: 'A' 'B' 'Test' 'Test'
ssp0 ... ssp99 gssp0 ... gssp99	Speichert den obersten Wert und entfernt ihn vom Zeichenfolgenstapel	1	stack: 'A' 'B' 'Test' ssp0	stack: 'A' 'B' ss0: 'Test'

AAO-spezifische RPN-Befehle

Befehl	Beschreibung	Beispiel
(CALL:scriptgroup-scriptname)	Ein anderes Skript aufrufen. Parameter können durch das Symbol „ “ getrennt hinzugefügt werden.	(CALL:Scripts-ACT_HEADING_TO_AP) (CALL:MyScripts-AScript param1 param2)
(EXEC:xxxx,params)	Starten ein Programm auf Ihrem Computer; Parameter sind optional. Sie können Daten dynamisch einfügen mit %1 bis %9 für numerische Werte %s1 bis %s9 für Zeichenfolgen	(EXEC:notepad.exe) öffnet den Editor (EXEC:notepad.exe,test.txt) öffnet die Datei (L:myvar1) (L:myvar2) (EXEC:app.exe,%1 %2) übermittelt die Werte der LVars als Parameter an das Programm
zusätzliche EXEC-Optionen	'explorer' ,steam:/ 'rungameid/1465360' scat (EXEC:%s1,%s2) Steam-App 1465360 starten (SnowRunner) ,explorer' 'shell:appsFolder\Microsoft.FlightSimulator_8wekyb3d8bbwe!App' (EXEC:%s1,%s2) MS Store-App (MSFS) starten. App-IDs für Steam- und Store-Apps lassen sich mit dem PowerShell-Befehl „get-StartApps“ ermitteln	
(EXECBAT:...)	Wie EXEC, jedoch für Batch-Dateien/Shell-Skripte. Es wird kein Anwendungsfenster erstellt	
(WINVAR:xxxx)	Fügt eine Windows-Umgebungsvariable als Zeichenfolge ein	(WINVAR:windir) => 'C:\Windows'
(>WINVAR:xxxx)	Eine Zeichenfolge einer Umgebungsvariablen zuweisen	'This is a test' (>WINVAR:mywinvar) %(A:PLANE ALTITUDE, feet)!0.4f! (>WINVAR:mywinvar)
(FOCUS:xxxx)	Richte den Windows-Fokus auf einen anderen Prozess, zum Beispiel um virtuelle Tastatureingaben an diesen zu senden. xxxx ist der Prozessname aus dem Task-Manager, ohne die Endung (.exe, ...)	Sende einen virtuellen Tastendruck an LNM und wechsele dann zurück zu MSFS: (FOCUS:littlenavmap) (SPLIT:100) (VKD:29-157-162) (VKD:56-184-164) (VKD:20-148-84) (VKU:20-148-84) (VKU:29-157-162) (VKU:56-184-164) (SPLIT:100)(FOCUS:flightsimulator)
(KILL:xxxx)	Einen laufenden Prozess beenden	(KILL:littlenavmap)
(SCRIPTFILE:xxx) (SCRIPTFILE:xxx nnn) (SCRIPTFILE:... CACHE) (SCRIPTFILE:... COMPILE)	Laden und führen Sie eine Datei mit RPN-Skripten aus. Die Datei muss im Klartextformat vorliegen und im Verzeichnis „Documents\LorbyAxisAndOhs Files\Scripts“ gespeichert sein. <ul style="list-style-type: none"> • „nnn“ kann eine Zeilennummer oder ein Label-Name sein. • Die Option „CACHE“ hält das Skript im Speicher. • „COMPILE“ behandelt die gesamte Skriptdatei so, als wäre sie ein einziges, mehrzeiliges Skript. 	(SCRIPTFILE:myscript.txt) (SCRIPTFILE:myfolder\myscript.txt) (SCRIPTFILE:myscript.txt 23) Öffnet die Datei und startet ab Zeile 23

Befehl	Beschreibung	Beispiel
(CHECKLIST:xxx) (CHECKLIST:xxx nnn) (CHECKLIST:.... CACHE)	Lade eine Skriptdatei vom Typ „RPN-Checkliste“ und führe sie aus.	(CHECKLIST:mychecklist.txt) (CHECKLIST:myfolder\mychecklist.txt) (CHECKLIST:myscript.txt 23) Öffnet die Datei und startet ab Zeile 23
(CONVERSATION:xxx) (CONVERSATION:xxx nnn) (CONVERSATION:.... CACHE)	Eine Skriptdatei vom Typ „RPN-Konversation“ laden und ausführen.	(CONVERSATION:myconversation.txt) (CONVERSATION:myfolder\myconversation.txt) (CONVERSATION:myscript.txt 23) Öffnet die Datei und startet ab Zeile 23
(WSHSCRIPTFILE:filename language startmethod CACHE ASYNC)	Lade eine Datei mit Windows Script Host-Code (z. B. jscript) und führe sie aus; CACHE und ASYNC sind optional; die Datei wird zwischengespeichert und/oder asynchron ausgeführt.	(WSHSCRIPTFILE:jstest.txt jscript AaoEntry CACHE) Lädt die Methode „AaoEntry“ aus der JScript-Datei, führt sie aus und speichert sie für die spätere Verwendung im Cache.
(GOTO:label)	Nur in Checklisten oder Unterhaltungen verfügbar: Springe zu der Zeile, die mit „:“ und der Bezeichnung beginnt.	(L:checkvar) 2 == if{ (GOTO:mymark) } ...RPN code.. :mymark
(GOLINE:xxx)	Nur in Skriptdateien verfügbar: Springe zu Zeile xxx	(L:checkvar) 2 == if{ (GOLINE:123) }
(GOSUB:xxx) / (RETURN)	Nur in Skriptdateien verfügbar: Springe zu der Zeile, die mit „:“ und der gewünschten Bezeichnung beginnt. Der Code nach dieser Bezeichnung muss in einer einzigen Zeile mit (RETURN) abgeschlossen werden, wodurch ein Sprung zurück zum aufrufenden Code erfolgt.	(L:checkvar) 2 == if{ (GOSUB:mymark) } ...RPN code.. :mymark ...RPN code (RETURN)
(END)	Nur in Skriptdateien verfügbar: Skript beenden	
(LVARPRN:xxxx)	Den Inhalt einer String-LVar als RPN-Code verarbeiten	variable containing code: (L:myvar, String) call: (LVARPRN:myvar)
(VKD:...) (VKU:...)	Ereignisse „Virtual Key Down“ und „Key Up“. Der RPN-Skript-Editor verfügt über einen speziellen Eingabedialog, um eine tatsächliche Tastaturaktion den Tastencodes zuzuordnen (Schaltfläche „Insert VKey“)	(VKD:44-172-89):(SPLIT:100):(VKU:44-172-89)·
(VJBD:dev btn) (VJBU:dev btn)	vJoy-Taste (Taste nach unten/oben) am Gerät	(VJBD:1 2) (VJBU:1 2)
(VJBD:dev pov dir) (VJBU:dev pov dir)	vJoy: Diskrete POV-Bewegung nach unten/oben in Richtung dir (0=N, 1=E, 2=S, 3=W) auf dem Gerät dev	(VJBD:1 1 3) (VJBU:1 1 3)

Befehl	Beschreibung	Beispiel
(VJAX:dev axis value)	Wert (0–32767) an die vJoy-Achse (X, Y, Z, RX, RY, RZ, SL0, SL1) des Geräts „dev“ senden	(VJAX:1 Y 16500)
(VJPV:dev pov value)	Wert (-1 – 36000) an vJoy Continuous POV auf dem Gerät „dev“ senden	(VJPV:1 3 4000)
(VIGBD:btn) (VIGBU:btn)	ViGEm-Taste nach unten/oben X,Y,A,B,Up,Down,Right,Left,RightShoulder, LeftShoulder,RightThumb,LeftThumb,Back,Start,Guide	(VIGBD:LeftShoulder) (VIGBU:LeftShoulder)
(VIGAX:axis value)	Wert (-32676 – 32767) an die ViGEm-Achse senden LeftThumbX,LeftThumbY,RightThumbX,RightThumbY	(VIGAX:LeftThumbY 6583)
(VIGSL:slider value)	Wert (0–256) an den ViGEm-Schieberegler senden LeftTrigger,RightTrigger	(VIGSL:LeftTrigger 94)
(VMOBD:btn) (VMOBU:btn)	Maustaste (links, Mitte, rechts) gedrückt/losgelassen	(VMOBD:Left) (VMOBU:Left) (VMOBD:Left) (VMOBU:Left) ist ein Doppelklick an der aktuellen Position
(VMOBD:btn x y) (VMOBU:btn x y)	Maustaste an der Bildschirmposition x,y	(VMOBD:Right 1000 500) (VMOBU:Right 1000 500)
(VMOXY:x y)	Bewegen Sie den Mauszeiger an die Bildschirmposition x,y	(VMOXY:1000 1000)
(VMOWH:val)	Mausrad drehen um val (1 Klick entspricht +/- 120)	(VMOWH:120)
(VMOWH:val x y)	Das Mausrad an der Bildschirmposition x,y drehen	(VMOWH:-120 000 500)
(VMOMV:x y)	Mausbewegung in x- und y-„Mickey“ „Ein Mickey ist die Strecke, die eine Maus zurücklegen muss, damit eine Bewegung registriert wird.“ (MSDN)	(VMOMV:48 0) Bewegt die Maus auf einem 4K-Monitor um etwa 100 Pixel nach rechts

Befehl	Beschreibung	Beispiel
(SOUND:xxxx) (SOUND:xxxx vvv) (SOUND:xxxx vvv bbb) (SOUND:xxxx vvv bbb dd) (SOUND:xxxx vvv bbb dd ms)	Spielt eine Audiodatei ab. Die Dateien müssen hier gespeichert sein: \\Dokumente\LorbyAxisAndOhs Files\Sounds vvv ist die Lautstärke (0 – 100) bbb ist die Balance (-100 – 100) dd ist die Geräte-ID (siehe „Extras -> Liste der Audiogeräte anzeigen“) ms ist die Dauer des Tons in Millisekunden	(SOUND:myfile.wav) (SOUND:myfile.wav 50)
(SOUNDLOOP:xxxx), (SOUNDLOOP:xxxx vvv) (SOUNDLOOP:xxxx vvv bbb) (SOUNDLOOP:xxxx vvv bbb dd) (SOUNDLOOP:xxxx vvv bbb dd ms)	Spielt eine Audiodatei in einer Schleife ab	(SOUNDLOOP:myfile.wav) (SOUNDLOOP:myfile.wav 50)
(SOUNDVOLUME:xxxx vvv) (RANDOMSOUNDVOLUME:xxxx vvv)	Die Lautstärke einer aktiven Audiodatei ändern (0 bis 100)	(SOUNDVOLUME:myfile.wav 75)
(SOUNDBALANCE:xxxx vvv) (RANDOMSOUNDBALANCE:xx xx vvv)	Die L/R-Balance einer aktiven Audiodatei ändern (-100 bis 100)	(SOUNDBALANCE:myfile.wav -50)
(STOPSOUND:xxxx)	Hält eine Audiodatei an	(STOPSOUND:myfile.wav)
(RANDOMSOUND:xxxx) (RANDOMSOUND:xxxx vvv) (RANDOMSOUND:xxxx vvv bbb) (RANDOMSOUND:xxxx vvv bbb dd)	Spielt eine zufällige Audiodatei aus einem Verzeichnis unter „Dokumente\LorbyAxisAndOhs Files\Sounds“ ab vvv ist die optionale Lautstärke (0–100)	(RANDOMSOUND:AtcChatter) (RANDOMSOUND:AtcChatter 50) unter Verwendung von Dateien aus LorbyAxisAndOhs Files\Sounds\AtcChatter*.*
(STOPRANDOMSOUND:xxxx)	Beendet die Wiedergabe des aktuell abgespielten Zufallsgeräuschs	(STOPRANDOMSOUND:AtcChatter)
(ONESHOT)	Füge dies am Ende deines Skripts ein, wenn es während eines Fluges nur einmal ausgeführt werden soll	(LISTEN_FOR_RPN:(A:PLANE·ALTITUDE,·Feet)·3000·>):if{ ·(SOUND:crew4_climbingdevice.wav)·}·(ONESHOT)

Befehl	Beschreibung	Beispiel
(SPEAK:xxxx)	Verwendet die Windows-Funktion „TextToSpeech“, um einen Text vorlesen zu lassen. Sie können Daten dynamisch einfügen mit ,%1‘ bis ,%9‘ für Zahlenwerte ,%s1‘ bis ,%s9‘ für Zeichenfolgen; fügen Sie ‚u‘/‘l‘ für Groß-/Kleinschreibung hinzu, fügen Sie ‚s‘ hinzu, um Zeichenfolgen im NATO-Alphabet zu buchstabieren	Dieses Skript gibt die aktuelle Höhe, Kurs und Geschwindigkeit an (A:INDICATED·ALTITUDE,·Feet)·flr·(A:PLANE·HEADING·DEGREES·GYRO,·Degrees)·flr·(A:AIRSPEED·INDICATED,·Knots)·flr·(SPEAK:We·are·at·%1·feet,·heading·%2·degrees·at·%3·knots) Buchstabieren eines ICAO-Codes: (A:GPS WP NEXT ID, String) (SPEAK:Next waypoint is %s1s)
(SPEAKTOFILE:ffff xxxx)	Speichere die Sprachausgabe von xxxx in der Datei Dokumente\\LorbyAxisAndOhsFiles\\Sounds\ffff	(SPEAKTOFILE:test.wav This is a test)
(VOICE:xxxx)	Ändert die Stimme der Windows-Funktion „Text-to-Speech“. Beachten Sie, dass die Stimmen lokalisiert sind; achten Sie darauf, eine Stimme auszuwählen, die die richtige Sprache spricht. Alle in Ihrem Windows-System installierten TTS-Stimmen, einschließlich ihrer Parameter, werden in diesem Dialogfeld angezeigt: „Extras->Liste aller TTS-Stimmen anzeigen“ Wenn Sie mit der rechten Maustaste auf einen Stimmennamen klicken, wird dieser in die Zwischenablage kopiert, sodass Sie ihn mit Strg+V in Ihr Skript einfügen können.	Dieses Skript spricht verschiedene Sätze mit unterschiedlichen Stimmen. Du kannst die Stimme, die Sprechgeschwindigkeit oder die Lautstärke erst ändern, wenn der vorherige Satz zu Ende ist. (VOICE:Microsoft·David·Desktop)·(SPEAK:Testing·the·TTS·feature)·(WAIT:3000)·(VOICE:Microsoft·Zira·Desktop)·(SPEAK:I·want·to·test·too)·(WAIT:3000)·(VOICE:Microsoft·Hedda·Desktop)·(SPEAK:Ich·will·Auch·testen)
(VOICERATE:xxxx)	Sprachgeschwindigkeit ändern, -10 bis 10	(VOICERATE:-10)
(VOICEVOLUME:xxxx)	Lautstärke von 0 bis 100 einstellen	(VOICEVOLUME:50)
(VOICEBALANCE:xxxx)	Balance links/rechts von -100 bis 100 einstellen	(VOICEBALANCE:-75)
(VOICEDEVICE:xxxx)	Ändern Sie die ID des Audiogeräts für die Sprachausgabe (siehe „Extras -> Liste der Audiogeräte anzeigen“)	(VOICEDEVICE:1)
(VOICEEFFECTS:aaa-bbb-ccc)	Füge der Stimme eine Effektkette hinzu (siehe Kapitel über Soundeffekte)	(VOICEEFFECTS:7,880,20-1,3-4,50,50)
(VOICEMIX:ffff)	Mische die Sprachausgabe mit der Datei Dokumente\\LorbyAxisAndOhsFiles\\Sounds\ffff	(VOICEMIX:helicotper.wav)
(SAVEFLIGHT:xxxx)	Speichert den aktuellen Flug in der Datei „xxxx“. Der gespeicherte Flug befindet sich im Dateiverzeichnis des Simulators unter \\Dokumente\	(SAVEFLIGHT:myflight)

Befehl	Beschreibung	Beispiel
(SPLIT:xxx), (SPLIT:xxx yyy)	Dieser Befehl teilt das Skript an dieser Stelle in zwei Teile und setzt die Verarbeitung des Codes fort, sobald die Zeit abgelaufen ist. Alle Werte, die vor dem SPLIT-Befehl auf dem RPN-Stack berechnet wurden, gehen verloren; alle Variablen werden erneut eingelesen. SPLIT kann nicht innerhalb von if/els-Anweisungen verwendet werden und akzeptiert nur feste Werte; %n-Parameter funktionieren hier nicht.	(SPLIT:1000): 1 Sekunde warten, dann den Rest des Codes als neues Skript ausführen. (SPLIT:2000 8000): Eine zufällige Zeit zwischen 2 und 8 Sekunden warten, dann den Rest des Codes als neues Skript ausführen.
(WAIT:xxx), (WAIT:xxx yyy)	hält die Ausführung des Skripts für die gewünschte Zeit an. Zu diesem Zeitpunkt sind alle Berechnungen bereits abgeschlossen; es handelt sich lediglich um eine Unterbrechung der Ausführung. Diese Funktion kann innerhalb von if/else-Strukturen verwendet werden.	(WAIT:1000): Verarbeitung für 1 Sekunde unterbrechen (WAIT:2000 8000): Eine zufällige Zeit zwischen 2 und 8 Sekunden abwarten, dann mit der Verarbeitung fortfahren
(READFLIGHTPLAN), (READFLIGHTPLAN:xxxx)	Dieser Befehl liest den aktuellen Flugplan oder eine Flugplandatei in eine Reihe interner LVars ein: (L:FPL_WP_CNT) (L:FPL_WP_ALT:n) (L:FPL_WP_LAT:n) (L:FPL_WP_LON:n) (L:FPL_WP_ICAO:n, String) n ist der Index eines Wegpunkts im Flugplan, beginnend mit 0; FPL_WP_CNT ist die Gesamtzahl der Wegpunkte	READFLIGHTPLAN): Aktuellen Flugplan laden (READFLIGHTPLAN:test.pln): Laden \\Documents\Microsoft Flight Simulator 2020 Files\test.pln Sie können mithilfe eines Makros dynamisch auf Wegpunktdaten zugreifen LVar 5 (>L:fppos) ... (L:FPL_WP_ALT:@fppos, feet)
(HTTPPOST:www.xxx.yy contentype zzzzzz)	Sende eine HTTP-POST-Anfrage an die URL „http://www.xxx.yy“ mit dem Inhaltstyp und dem Inhalt zzzzzz	(HTTPPOST:localhost:8083/graphql application/json {"query":"mutation (\$direction: Int!) {dataRef {writeInt(name: \"groundservice.pushback\", value: \$direction)}}\", \"variables\": {\"direction\":1}}) Dieser Befehl bewirkt, dass der Fenix Airbus Pushback-Schlepper nach links abbiegt (Richtung 2 = rechts, 0 = geradeaus)
(HTTPSPOST:www.xxx.yy contentype zzzzzz)	Wie HTTPPOST, jedoch für https://...	
(OPENURL:url)	Öffnen Sie eine Webseite in Ihrem Standardbrowser. Wenn die URL nicht mit „http“ beginnt, wird davon ausgegangen, dass es sich um einen relativen Pfad zu einer AAO-Webseite handelt.	(OPENURL:a320cdu/index.html) (OPENURL:http://www.google.com)

Befehl	Beschreibung	Beispiel
(QUERYJSON:url user password) (QUERYXML:url user password)	Abfrage eines Webdienstes, der entweder JSON oder XML zurückgibt. Das Ergebnis wird in LVars vom Typ String geschrieben. Benutzername und Passwort sind optionale Parameter.	(QUERYXML:https://aviationweather-cprk.ncep.noaa.gov/adds/dataserver_current/httpparam?dataSource=metars&requestType=retrieve&format=xml&stationString=EDDM&hoursBeforeNow=1)
(DOWNLOADJSON:url filename user password) (DOWNLOADXML:url filename user password)	Wie oben, jedoch werden die Ergebnisse statt in LVars in eine Datei geschrieben (absoluter Pfad oder relativ zu \Documents\LorbyAxisAndOhs Files\)	(L:response.data.METAR.raw_text, String):'EDDM 301850Z AUTO 07004KT 030V100 CAVOK 17/09 Q1020 NOSIG'
(SENDMAIL:somebody@domain.com subject body)	Senden Sie eine E-Mail mit Betreff und Text an den Empfänger. Denken Sie daran, Ihren E-Mail-Server im Menü „Extras“ zu konfigurieren!	(SENDMAIL:me@gmx.de Greetings from AAO This is a test message from AAO)
(LOAD_SIMBRIEF:xxx) (LOAD_SIMBRIEF_PLAN:xxx)	Lade die OFP-Daten des Benutzers xxx. Dazu gehören die Tokens „general“, „origin“, „destination“, „alternate“, „aircraft“, „fuel“, „times“, „weights“ und „atc“. Jede Eigenschaft wird in eine LVar (numerisch oder als Zeichenkette) geschrieben, der Flugplan wird in den simulationsinternen Flugplan geschrieben.	(L:simbrief.origin.icao_code, String) (L:simbrief.destination.trans_alt) etc.
(LOAD_SIMBRIEF) (LOAD_SIMBRIEF_PLAN)	Wie oben, jedoch unter Verwendung des SimBrief-Benutzernamens, den Sie im AAO-Menü unter „Extras“ eingeben können	
(AAO_ONLINE_MODE)	AAO-Verbindung zum Simulator aktivieren/deaktivieren	
(AAO_OFFLINE_MODE:xxxx)	Offline-Modus aktivieren, Konfiguration xxx laden	(AAO_OFFLINE_MODE:Cessna 152 Asobo)
(AAO_RECONNECT)	Vom Simulator trennen und dann wieder verbinden	
(AAO_RESET_CONFIG)	Aktuelle Flugzeugkonfiguration neu laden	
(AAO_RESCAN_DEVICES)	Alle Eingabegeräte erneut scannen	
(LOADDESKLAYOUT:xxxxx)	Desktop-FIPs/Anzeigenlayout mit dem Namen „xxxxx“ laden	(LOADDESKLAYOUT:Cessna Analog Gauges)
(LOADWEBLAYOUT:xxxxx)	Web-FIP-Layout mit dem Namen „xxxxx“ laden	(LOADWEBLAYOUT:Cessna Analog Gauges)
(LOADDESKWEBLAYOUT:xxx)	Lade das App-/Web-Layout mit dem Namen xxx	(LOADDESKWEBLAYOUT:Support Apps)
(AAO_SHOW_x:n) (AAO_HIDE_x:n)	Zeige/verberge die Anzeige vom Typ x und mit dem Index n x ist DESKFIP, WEBFIP, APPWEB n ist 0 bis (Anzahl der FIPs im Layout minus 1)	(AAO_HIDE_DESKFIP:2) blendet den Desktop-FIP aus, der sich an dritter Stelle der Liste im Handler-Dialog befindet

Befehl	Beschreibung	Beispiel
(AAO_SAVE_DESKFIP:n lvar) (AAO_SAVE_WEBFIP:n lvar)	Die unter Index n angezeigte Anzeige als Base64 in (L:lvar, String)	
(RENDERGAUGE:name lvar)	Die Messgrößen-Definition mit „name“ als Base64 in (L:lvar, String)	
(UNLISTEN:xxx)	Beende einen „LISTEN_FOR“-Befehl. XXX ist entweder die im Skript-Header angegebene ID oder das K-Ereignis	
(REGEXSPLIT:text regex case/nocase arrayname)	Dadurch wird ein Text anhand eines regulären Ausdrucks (mit oder ohne Groß-/Kleinschreibung) aufgeteilt und das Ergebnis in einem Array gespeichert. Verwende in deinem regulären Ausdruck „&pipe;“ anstelle des Zeichens „ “.	'testcase·atis·with·A·information'·(REGEXSPLIT:%s1 testcase·atis·with·([A-Z])·information case ATISARRAY)
(HIDWRITE:vid,pid,value)	Schreibe den Wert in den Feature-Puffer des HID-Geräts mit PID und VID. Der Wert kann im numerischen oder hexadezimalen Zeichenfolgenformat vorliegen. 0x11223344 entspricht „11223344“.	(HIDWRITE:0x294B,0x1901,'00008000') Setze das Bit ganz links im zweiten Byte des Feature-Puffers, oder ein Honeycomb Bravo (Warnleuchte für niedrigen Hydraulikdruck) und setzt den Rest auf AUS
(HIDREAD:vid,pid,lvar) (only with specific devices)	Liest den Feature-Puffer des HID-Geräts mit PID und VID in die Variable „lvar“ ein. Gib lediglich den Namen der Variable „lvar“ an.	(HIDREAD:0x294B,0x1901,hidtest) Liest den Inhalt des Honeycomb Bravo (= den Status der Anzeigen) in die Variable (L:hidtest, String) ein
(HIDWRITEOR:vid,pid,val) (only with specific devices)	Erweitert den Funktionspuffer des HID-Geräts mithilfe des bitweisen Operators ; berücksichtigt dabei den vorhandenen Puffer. Kann beispielsweise verwendet werden, um einzelne Bits im Puffer auf 1 zu setzen	'0080000000' (HIDWRITEOR:0x294B,0x1901,'%s1') schaltet die Autopilot-Anzeige am Honeycomb Bravo ein, lässt alle anderen Anzeigen unverändert
(HIDWRITEAND:vid,pid,val) (only with specific devices)	Erweitert den Feature-Puffer des HID-Geräts mithilfe des Bitoperators &; berücksichtigt dabei den vorhandenen Puffer. Kann beispielsweise verwendet werden, um einzelne Bits im Puffer auf 0 zu setzen	0x80000000 ~ (HIDWRITEAND:0x294B,0x1901,%1) schaltet die Autopilot-Anzeige am Honeycomb Bravo aus, lässt alle anderen Anzeigen unverändert
(HIDDEVICES)	Schreibt eine Liste aller HID-Geräte mit VID und PID in die Datei \Documents\LorbyAxisAndOhsFiles\HID_Devices_List.txt	
(GETBUTTONS:guid lvarname)	Schreibt eine CSV-Zeichenkette mit den aktuellen Tastenzuständen des Spielecontrollers zusammen mit der „GUID“ in (L:lvarname,	(GETBUTTONS:4efea320-6a76-11ee-8001-444553540000 btnlst)

	String) Die GUID erhältst du, indem du in der linken Liste der Dialoge „Hardware-Änderung“ und „Geräte-Blacklist“ mit der rechten Maustaste auf den Controller klickst – dadurch wird die GUID in die Windows-Zwischenablage kopiert, und du kannst sie in das Skript einfügen	(L:btnlst, String) is "0,0,1,0,-1" = 4 Tasten und ein POV-Regler. Diese Liste enthält so viele Tasten und POVs, wie das Gerät aufweist.
--	---	--

Befehl	Beschreibung	Beispiel
(GETAXIS:guid lvarname)	Schreibt eine CSV-Zeichenkette der aktuellen Achsenwerte des Spielecontrollers mit der Kennung „guid“ in (L:lvarname, String)	(GETAXIS:4efea320-6a76-11ee-8001-444553540000 axilst)· (L:axilst, String) ist X,Y,Z,Rx,Ry,Rz,sl1,sl2
(MSFS_ACTIVATE_ADDON:) (MSFS_DEACTIVATE_ADDON:)	Aktivieren oder deaktivieren Sie ein Add-on-Paket in den Ordnern „MSFS Community“ oder „Official“, indem Sie die Datei „manifest.json“ umbenennen. Geben Sie den Namen des Paketordners so an, wie er unter \Community oder \Official\Steam / \Official\OneStore erscheint. Der Ordnername kann auch vollständig angegeben werden (beginnend mit dem Laufwerksbuchstaben), falls AAO den Speicherort des MSFS-Paketordners in der Registrierung nicht finden kann.	(MSFS_ACTIVATE_ADDON:lorbyasi-atc-bridge) (MSFS_DEACTIVATE_ADDON:asobo-aircraft-a320-neo)
(DICTATION_START:engine language)	Startet den Diktatmodus. „engine“ kann einer der folgenden Werte sein: „Windows“, „Azure“ oder „Cortana“. Die Sprache ist eine Länderkennung (en-US, de-DE usw.).	(DICTATION_START:Windows en-US)
(DICTATION_STOP) (DICTATION_END)	Will stop the dictation mode. "END" will also terminate the dictation engine, use this when changing language	
(DICTATION_STOP_AND_PARSE) (DICTATION_END_AND_PARSE)	Beendet den Diktatmodus und sendet das Ergebnis zur Verarbeitung an das Spracherkennungsmodul.	
(SIMVAL:rpncode)	SIMVAL übermittelt den RPN-Code zur Verarbeitung an die SIM-Karte und gibt das Ergebnis der Berechnung zurück. Dies kann in allen Fällen genutzt werden, in denen SimConnect den Wert nicht abrufen könnte. Beispiel: Variable zur Abfrage des elektrischen Busses	(SIMVAL:16·(>A:BUS·LOOKUP·INDEX,·Number)·(A:BUS·CONNECTION·ON:15,·Bool))·1·==·:if{...}
(LOG:text)	Schreibe den Text in die Protokolldatei unter \Dokumente\LorbyAxisAndOhs Files\	

--	--	--

Windows-Operationen

Mit diesen Befehlen können Sie auf einige Funktionen der Win32-API zugreifen.

Auf alle Fenster kann über ihren Titel zugegriffen werden, entweder vollständig oder teilweise (ab dem Anfang). Wenn es mehr als ein Fenster mit demselben Namen gibt, fügen Sie einen Index mit „~“ an.

Alle Parameter sind numerisch; Literale können in hexadezimaler Darstellung wie „0x00C00000“ angegeben werden.

Befehl	Beschreibung
(WINDOWSTYLESET:title~index style)	Legen Sie den Stil des Fensters fest. Fenster-Stile
(WINDOWSTYLEGET:title~index lvarname)	Den Stil eines Fensters abrufen und in L:lvarname schreiben)
(WINDOWSTYLEADD:title~index style)	Den Stil einem Fenster hinzufügen (style & toremove)
(WINDOWSTYLESUB:title~index style)	Den Stil aus einem Fenster entfernen (style & ~toremove)
(WINDOWEXSTYLE (set, get, add, sub)	Für die erweiterten Stile gilt dieselbe Logik wie oben Erweiterte Windows-Stile
(WINDOWSHOW:title~index nCmdShow)	Ein Fenster mithilfe des numerischen Befehlscodes anzeigen ShowWindow-Befehle
(WINDOWPOSITIONGET:title~index x y w h)	Ermittle die Position und Größe eines Fensters und speichere sie unter (L:x), (L:y), (L:w), (L:h)
(WINDOWPOSITIONSET:title~index x y w h)	Wähle ein Fenster aus und lege Position und Größe fest (x, y, Breite, Höhe)

Befehl	Beschreibung
(WINDOWTITLESET:title~index header)	Ändere die Kopfzeile des Fensters, beginnend mit dem Titel
(WINDOWTITLEGET:title~index lvarname)	Die Kopfzeile eines Fensters abrufen und in (L:lvarname, String) schreiben
(WINDOWBORDERLESS:title~index)	Ein Fenster randlos machen
(WINDOWCLOSE:title~index)	Versuchen das externe Fenster zu schließen (nicht immer möglich/zulässig)

Dateisystemoperationen

Wenn der Dateiname in einem Befehl nicht vollständig angegeben ist (C:\...), wird davon ausgegangen, dass der Speicherort relativ zu \Dokumente\LorbyAxisAndOhs Files ist.

Befehl	Beschreibung	Beispiel
(LOADTEXTFILE:filename arrname cap)	Lädt eine Textdatei „filename“ in ein STRARR „arrname“ mit der Kapazität „cap“	(LOADTEXTFILE:C:\mytextfile.txt MY_MENU 6) lädt die ersten 6 Zeilen der Textdatei in das Array MY_MENU (STRARR:MY_MENU:6) 0 get ergibt die erste Zeile, (STRARR:MY_MENU:6) 1 get die zweite usw.
(LOADTEXTFILE:filename var)	Lädt den Inhalt der Datei in eine einzelne LVar vom Typ String	(LOADTEXTFILE:C:\mytextfile.txt L:myvar) loads the entire text into (L:myvar, String)
(LOADNUMFILE:filename arrname cap)	Lädt eine Textdatei „filename“, in der jede Zeile eine Zahl ist (z. B. 32123.4323), in ein NUMARR-Array „arrname“ mit der Kapazität „cap“	(LOADNUMFILE:C:\mytextfile.txt MY_VALUES 6) lädt die ersten 6 Zeilen der Textdatei als Zahlen in das Array MY_MENU (NUMARR:MY_VALUES:6) 0 get gibt die Zahl aus der ersten Zeile zurück (NUMARR:MY_VALUES:6) 1 get die zweite usw.
(WRITEFILE:filename arrname APPEND)	Schreibt das Array mit dem Namen „arrname“ in die Textdatei „filename“. APPEND ist optional; wird es weggelassen, wird die Datei überschrieben.	(WRITEFILE:C:\mytextfile.txt MY_MENU) überschreibt die Textdatei mit dem Inhalt des STRARR (WRITEFILE:C:\mynumfile.txt MY_VALUES APPEND) fügt den Inhalt des NUMARR an die Datei an
(DELETEFILE:filename)	Lösche die Datei „filename“	
(COPYFILE:sourcefile targetfile)	Kopiert die Quelldatei (sourcefile) in die Zieldatei (targetfile)	
(MOVEFILE:sourcefile targetfile)	Verschiebt die Quelldatei(sourcefile) in die Zieldatei (targetfile)	
(DIRLIST:dirname arrname cap)	Lese die Dateinamen aus dem Verzeichnis „dirname“ in das Array „arrname“ mit der Kapazität „cap“ (absolut oder relativ zum Pfad der AAO-Dokumente)	(DIRLIST:MyGauge mystrarr 50)

Befehl	Beschreibung	Beispiel
(LOADIMAGE:filename varname)	Lädt eine Binärbilddatei im Base64-Format in eine String-Lvar	(LOADIMAGE:C:\animage.png myimage) => (L:myimage, String) enthält das Base64-kodierte Bild
(LOADIMAGE:filename varname col1 col2)	Lädt das Bild und ersetzt die Farbe „col1“ durch „col2“. Farben werden als HTML-Hex-Farbcodes im Format #RRGGBB oder #RRGGBBAA angegeben.	(LOADIMAGE:UserGauges\1024\McpEncoderKnob.png picture1 #767676 #0000FF) => (L:Bild1, String) enthält den Base64-Code; Mittelgrau wurde durch Blau ersetzt
(WRITEIMAGE:filename varname)	Einen Base64-String in eine Binärbilddatei schreiben	(WRITEIMAGE:C:\animage.png myimage) => (L:myimage, String) wird in eine Binärdatei geschrieben
(REPLACECOLOR:varname col1 col2)	Eine Farbe in einer Base64-Bildzeichenfolgenvariablen ersetzen. Farben sind HTML-Hex-Farbcodes im Format #RRGGBB oder #RRGGBBAA	(REPLACECOLOR:picture1 #8A8BB066 #FF000044)
(EXPORTVARS:filename)	Exportiert alle Variablen, die sich derzeit im AAO-Cache befinden, in die Datei „Dateiname“. Das Dateiformat ist Text/TSV (tabulatorgetrennte Werte).	
(REPLACEVARS:sourcefile,targetfile)	Liest die Quelldatei und ersetzt alle Variablenzeichenfolgen darin durch ihre tatsächlichen Werte. Das Ergebnis wird in die Zieldatei geschrieben. Die Quelldatei muss im Textformat vorliegen	
(EXPORTAI:filename)	Schreibt Daten zu allen derzeit in der Simulation vorhandenen KI-Objekten in die Datei „Dateiname“. Das Dateiformat ist Text/TSV (tab-separierte Werte).	

Dynamische Parameter

Sie können dynamische Parameter in den AAO-Befehlen verwenden

- Fügen Sie %1 bis %9 ein, um Werte aus dem Zahlenstapel auszulesen
- Fügen Sie %s1 bis %s9 ein, um Werte aus dem Zeichenfolgenstapel auszulesen

Beispiele:

300 (WAIT:%1) entspricht (WAIT:300)

'Left' 500 200 (VMOBD:%s1|%1|%2) entspricht (VMOBD:Left|500|200)

AAO-spezifische Skript-Header

Skript-Header müssen am Anfang eines Skripts stehen, da sie über die weitere Verarbeitung entscheiden

Header	Beschreibung
(WSH:language startmethod ASYNCR)	Verwenden Sie den Windows Script Host mit der gewünschten Sprache, um den Code zu verarbeiten. Mit dem optionalen Parameter ASYNCR wird das Skript asynchron (=parallel) verarbeitet. (WSH:jscript AaoEntry) ... (WSH:vbscript AaoEntry ASYNCR) ...
(SIMPROC)	Leitet das Skript zur Verarbeitung an den Simulator weiter. Dies kann bei ausgefallener RPN-Logik hilfreich sein, die von SimConnect nicht direkt unterstützt wird
(LISTEN_FOR_K:xxxx) scriptcode	Dadurch wird AAO angewiesen, auf eine bestimmte Ereignis-ID (K-Events) zu achten Beispiel: (LISTEN_FOR_K:STROBES_TOGGLE) (SOUND:strobestoggled.wav) spielt eine Audiodatei ab, sobald das Ereignis K:STROBES_TOGGLE empfangen wird
(LISTEN_FOR_VOICE:xxxx) scriptcode (LISTEN_FOR_VOICE:xxxx id)	Mit diesem Skript-Header wird der Skriptcode ausgeführt, sobald AAO den Sprachbefehl empfängt. Mehrere Sprachbefehle können mit dem Symbol „ “ gruppiert werden, oder Sie können auch eine SRGS-Datei verwenden. Fügen Sie „(ONESHOT)“ am Ende des Skriptcodes hinzu, um daraus ein einmaliges Ereignis zu machen. Die ID dient für den Befehl „UNLISTEN“. (LISTEN_FOR_VOICE:mayday help) (SPEAK:What is the problem?)
(LISTEN_FOR_RPN:evalcode) scriptcode (LISTEN_FOR_RPN:evalcode id) (REPEAT_WHILE_RPN:evalcode) scriptcode	Dadurch wird der Skriptcode aufgerufen, sobald eine bestimmte Bedingung erfüllt ist – „evalcode“ ist ein RPN-Skript, das entweder 0 (falsch) oder 1 (wahr) zurückgeben muss. Das Ergebnis wird an das Skript weitergeleitet. Füge „(ONESHOT)“ am Ende des Skriptcodes hinzu, um es zu einem einmaligen Ereignis zu machen. Die ID dient für den Befehl UNLISTEN. (LISTEN_FOR_RPN:(A:LIGHT STROBE, Bool) 1 ==) if{ (SPEAK:Strobes are on) (ONESHOT) }
[rpn_condition] rest of script	Wenn ein Skript mit einem RPN-Bedingungsausdruck in [] beginnt, wartet AAO, bis diese Bedingung erfüllt ist, bevor der Rest des Skripts ausgeführt wird. Im Gegensatz zu LISTEN_FOR_RPN wird dieses Skript nur einmal ausgeführt. [(A:AUTOPILOT·HEADING·LOCK·DIR,·Degrees)·flr·(A:PLANE·HEADING·DEGREES·GYRO,·Degrees)·flr·==]·(SPEAK:Heading reached)

RPN-Wert-Arrays

In AAO-Skripten können Sie Arrays mit numerischen Werten oder Zeichenfolgen verwenden. Sie werden ähnlich wie normale Variablen verwendet, mit folgender Syntax:

(NUMARR:<arrayname>:<capacity>)

bezieht sich auf ein Array mit numerischen Werten. Beispiel: (NUMARR:mynumarray:5) ist ein Array mit einer Kapazität von 5 numerischen Werten.

(STRARR:<arrayname>:<capacity>)

bezieht sich auf ein Array von Zeichenfolgen. Beispiel: (STRARR:mystrarray:8) ist ein Array mit 8 Zeichenfolgen.

Die Kapazität ist eine feste Grenze. Wenn die Anzahl der Werte die Kapazität überschreitet, wird der erste Wert (Index 0) aus dem Array entfernt.

1 (NUMARR:myarr:4) push 3 (NUMARR:myarr:4) push 5 (NUMARR:myarr:4) push 7 (NUMARR:myarr:4) push [1,3,5,7]

9 (NUMARR:myarr:4) push [3,5,7,9]

(NUMARR:myarr:4) sum [24]

(NUMARR:myarr:4) 2 get [7]

12 (NUMARR:myarr:4) 2 set [3,5,12,9]

(NUMARR:myarr:4) sum [29]

(NUMARR:myarr:4) avg [7.25]

Dynamische Listen in RPN

Dynamische Listen ähneln Arrays, haben jedoch keine festgelegte Kapazität. Sie können beliebig viele Werte in die Liste einfügen.

(NUMARR:<arrayname>)

verweist auf eine dynamische Liste numerischer Werte. Beispiel: (NUMARR:mynumlist)

(STRARR:<arrayname>:)

verweist auf eine dynamische Liste von Zeichenfolgen. Beispiel: (STRARR:mystrarray)

Mit dem RPN-Operator „next“ können Sie eine Liste durchlaufen.

1 (NUMARR:myarr) push 3 (NUMARR:myarr) push 5 (NUMARR:myarr) push 7 (NUMARR:myarr) push [1,3,5,7]

9 (NUMARR:myarr) push [1,3,5,7,9]

(NUMARR:myarr) sum [25]

(NUMARR:myarr) 2 get [5]

12 (NUMARR:myarr) 2 set [1,3,12,7,9]

(NUMARR:myarr) sum [32]

(NUMARR:myarr) avg [6.4]

Die folgenden Operatoren können mit Arrays verwendet werden:

Befehl	Operation	Argumente	Beispiel
push	Füge einen Wert am Ende des Arrays hinzu	1	12.5 (NUMARR:myarr:4) push 'hello' (STRARR:myarr:4) push
pop	Entfernt den letzten Wert aus dem Array und legt ihn auf den aktuellen Verarbeitungstapel	0	(NUMARR:myarr:4) pop (STRARR:myarr:4) pop
peek	Liest den letzten Wert des Arrays aus, ohne ihn zu entfernen, und legt ihn auf den aktuellen Stapel	0	(NUMARR:myarr:4) peek (STRARR:myarr:4) peek
clear	Alle Werte im Array löschen	0	(STRARR:myarr:4) clear
get	Den Wert an einem bestimmten Index des Arrays abrufen, sofern er vorhanden ist; der Index reicht von 0 bis zur aktuellen Anzahl der Werte	1	(NUMARR:myarr:4) 2 get (STRARR:myarr:4) 1 get
set	Den Wert an einer bestimmten Position im Array ersetzen; der Index reicht von 0 bis zur Gesamtanzahl der Elemente	2	12.5 (NUMARR:myarr:4) 2 set '12.5' (STRARR:myarr:4) 1 set
rmv	Den Eintrag an der Stelle entfernen	1	(NUMARR:myarr) 2 rmv
ccap	Die Größe eines Arrays/einer Liste ändern	1	(NUMARR:myarr) 30 ccap
sum	Gibt die Summe aller Werte (numerisch) oder eine verkettete Zeichenfolge aller Werte (String) zurück	0	(NUMARR:myarr:4) sum (STRARR:myarr:4) sum
avg	Gibt den Durchschnitt aller Werte zurück (nur NUMARR)	0	(NUMARR:myarr:4) avg
count	Gibt die Anzahl der Werte im Array zurück	0	(NUMARR:myarr:4) count
cnt	Prüfe, ob ein Array einen Wert enthält, und gib den Index zurück. Bei Zeichenfolgen-Arrays reicht bereits eine übereinstimmende Teilzeichenfolge aus.	1	'12345' (STRARR:myarr:4) 2 set '23' (STRARR:myarr:4) cnt ergibt 2
next	Den nächsten Wert in einem Array zurückgeben	0	(STRARR:myarr) next (>L:currentvalue, String)
rset	Den Listenzähler (für „next“) auf 0 zurücksetzen	0	(STRARR:myarr) rset
revs	Das Array umkehren	0	(STRARR:myarr) revs
sora, sord	Sortiert das Array aufsteigend oder absteigend	0	(STRARR:myarr) sora
tocsv	Gibt eine Zeichenkette zurück, in der alle Werte durch ein Zeichen getrennt sind	1	' ' (STRARR:myarr) tocsv -> 'first second third'

RPN-HashMaps

HashMaps verknüpfen ein String-Literal Ihrer Wahl mit einem anderen Literal oder mit einer LVar, einem Array oder einer dynamischen Liste.

Beispiel 1: Speichern einfacher Werte in einer HashMap:

```
'one' 'item1' (HASHMAP:mymap) mset
'two' 'item2' (HASHMAP:mymap) mset

'item1' (HASHMAP:mymap) mget => 'one'
'item2' (HASHMAP:mymap) mget => 'two'

'three' 'item1' (HASHMAP:mymap) mset

'item1' (HASHMAP:mymap) mget => 'three'
```

Beispiel 2: Festlegen der Werte zweier Lvars und Speichern der Referenz auf die LVar in der HashMap:

```
(HASHMAP:mymap) mclr
'Test' (>L:tester, String) 25 (>L:ntester)
'L:tester, String' 'strvar' (HASHMAP:mymap) mset
'L:ntester' 'numvar' (HASHMAP:mymap) mset
```

Abrufen von Lvar-Werten

```
'strvar' (HASHMAP:mymap) mget => 'Test'
```

```
'numvar' (HASHMAP:mymap) mget => 25
```

In diesem Fall speichert die HashMap lediglich einen Verweis. Wenn der Wert der LVar geändert wird, wird beim Abrufen der Variablen aus der HashMap ebenfalls der neue Wert zurückgegeben.

```
(L:ntester) 10 + (>L:ntester)
```

```
'numvar' (HASHMAP:mymap) mget => 35
```

Anstelle von L: Variables können Sie auch die Elemente STRARR und NUMARR verwenden.

Die folgenden Operatoren können mit HashMaps verwendet werden:

Befehl	Operation	Argumente	Beispiel
mset	Fügt einen Wert am Ende des Arrays hinzu	2	'one' 'item1' (HASHMAP:mymap) mset
mget	Lies den Wert, auf den der Schlüssel in der Map verweist, aus und lege ihn auf den aktuellen Verarbeitungstapel	1	'item1' (HASHMAP:mymap) mget
mrem	Den Eintrag entfernen, auf den der Schlüssel verweist	1	'item1' (HASHMAP:mymap) mrem
mclr	Alle Einträge in der Liste löschen	0	(HASHMAP:mymap) mclr
mexist	Prüft, ob ein Schlüssel in der Map vorhanden ist, und gibt 0 oder 1 zurück	1	'item1' (HASHMAP:mymap) mexist

Umgang mit SimObjekten mittels RPN-Skripten

AAO RPN verfügt über Befehle, mit denen simulierte Objekte verwaltet werden können.

Ein simuliertes Objekt erstellen:

(CREATEOBJECT:title,REL|ABS,x,y,z,p,b,h,speed,onground,freeze,varname)

- title: ist der Teil „title=“ aus der sim.cfg eines SimObjekts.
- REL|ABS: gibt die Art der Objektpositionierung an, entweder RELativ zu Ihrer Position oder an einer ABSoluten Breiten-/Längenkoordinate
- x,y,z: Bei Verwendung von REL sind dies „Peilung, Entfernung, Höhenversatz“ relativ zu Ihrer eigenen Position. Bei Verwendung von ABS sind dies „Breitengrad, Längengrad, Höhe“
- p,b,h sind Neigung, Schräglage und Kurs des erstellten Objekts. Der Kurs ist ebenfalls relativ oder absolut.
- speed ist die Anfangsgeschwindigkeit des Objekts in Knoten.
- onground ist 0/1 und teilt der Simulation mit, ob das Objekt auf dem Boden erstellt werden soll oder nicht.
- freeze ist 0/1 und bewirkt, wenn gesetzt, dass das Objekt in seiner Ausgangsposition eingefroren wird.
- varname ist ein Lvar-Name Ihrer Wahl, um die ObjectID des Simobjekts zu speichern, sobald es erstellt wurde.

Beispiel:

```
(CREATEOBJECT:AfricanGiraffe,REL,45,400,0,0,0,45,0,1,0,objid_1)
```

Dadurch wird eine Giraffe erstellt, die sich 45 Grad rechts von dir befindet, 400 Fuß entfernt ist, 45 Grad von deiner eigenen Flugrichtung abweicht, auf dem Boden steht und deren Objekt-ID in der Variablen (L:objid_1) gespeichert wird

Zum Erstellen von Flugzeugen verwenden Sie

```
(CREATEOBJECT:title,tailnum,REL|ABS,x,y,z,p,b,h,speed,onground,freeze,varname)
```

Beispiel:

```
(CREATENONATC:PMDG 777-300ER PMDG House,aao,REL,90,200,0,0,0,0,0,1,1,objid_3)
```

Dadurch wird im Simulator ein KI-Flugzeug mit dem Rufzeichen „aao“ erstellt.

Ein geparktes KI-Flugzeug an einem Flughafen erstellen

```
(CREATEPARKED:title,tailnum,icao,varname)
```

Beispiel:

```
(CREATEPARKED:Generic Airliner Twin Engines Asobo 00,aao,eddf,objid_4)
```

Erstellen eines AI-Flugzeugs auf der Strecke gemäß einem Flugplan

(CREATEENROUTE:title,tailnum,flightnum,fpfile,fpposition,tng,varname)

- flightnum muss eine Zahl sein
- fpfile ist der Name einer .pln-Datei im Verzeichnis „Documents\LorbyAxisAndOhs Files\Flightplans“, die den Flugplan für die KI enthält.
- fpposition ist der Index und die Entfernung vom Wegpunkt des Flugplans, an dem das Flugzeug erstellt werden soll. 0,0 bedeutet, dass es zu Beginn des Fluges geparkt ist, 2,5 bedeutet die Mitte zwischen den Wegpunkten 2 und 3.
- tng wird auf 1 gesetzt, wenn das Flugzeug einen Touch-and-Go-Anflug durchführt, andernfalls auf 0.

Beispiel:

(CREATEENROUTE:Cessna Skyhawk G1000 Asobo, aao, 2348, IFR_edma_eddb.pln, 0.0, 0, objid_6)
mit der Flugplan-Datei namens „IFR_edma_eddb.pln“.

Ereignisse an ein KI-Flugzeug senden

Mit (SENDKEVENT:id,evt,val) können Sie K: Simulator-Ereignisse an KI-Flugzeuge senden.

Beispiel, basierend auf dem obigen CREATEENROUTE:

(L:idvar) (SENDKEVENT:%1, TOGGLE_NAV_LIGHTS, 0)

(L:idvar) (SENDKEVENT:%1, TOGGLE_BEACON_LIGHTS, 0)

(L:idvar) (SENDKEVENT:%1, GEAR_DOWN, 1)

Daten aus KI-Objekten auslesen

(GETOBJDATA:id,strarr,numarr)

(GETOBJSTRINGDATA:id,strarr,strarr)

Mit diesen Methoden können Sie Daten aus einem KI-Objekt auslesen, beispielsweise aus einem Flugzeug.

Die auszulesenden Variablen werden in STRARR bereitgestellt, die Ergebnisse werden entweder in ein numerisches Array oder ein String-Array geschrieben.

Jedes Flugzeug kann jeweils nur einen aktiven GET-Aufruf pro Typ haben

Beispiel: Abfrage des geografischen Standorts eines bei LIEO erstellten Flugzeugs und seines Titels.

```
(CREATEOBJECT:PMDG 777-200LR PMDG House (N772LR | 2020),ABS,40.892,9.506,50,0,0,90,0,0,1,idvar)
(SPLIT:15000)
(STRARR:get777) clear
(STRARR:gets777) clear
(NUMARR:data777) clear
(STRARR:datas777) clear
'A:PLANE LATITUDE, Degrees' (STRARR:get777) push
'A:PLANE LONGITUDE, Degrees' (STRARR:get777) push
(GETOBJDATA:@idvar,get777,data777)
'A:TITLE, String' (STRARR:gets777) push
(GETOBJSTRINGDATA:@idvar,gets777,datas777)
```

Daten in AI-Objekte schreiben

(SETOBJDATA:id|varname|value)

(SETOBJSTRINGDATA:id|varname|value)

Mit diesen Methoden können Sie Daten in ein AI-Objekt schreiben.

Beispiel: Erstellen eines Flugzeugs bei LIEO und Bewegen nach links und rechts

```
(CREATEOBJECT:PMDG 777-200LR PMDG House (N772LR | 2020),ABS,40.892,9.506,50,0,0,90,0,0,1,idvar)
(SPLIT:15000)
(SETOBJDATA:@idvar|A:PLANE LATITUDE, Degrees|40.892005)
(SPLIT:1000)
(SETOBJDATA:@idvar|A:PLANE LATITUDE, Degrees|40.892)
(SPLIT:1000)
(SETOBJDATA:@idvar|A:PLANE LATITUDE, Degrees|40.892005)
(SPLIT:1000)
(SETOBJDATA:@idvar|A:PLANE LATITUDE, Degrees|40.892)
(SPLIT:15000)
(REMOVEOBJECT:@idvar)
```

Alle AI-Objekte aus der Simulation abrufen

Mit (FETCHAI) können Sie alle aktuellen AI-Objekte in das STRARR „aidata“ einlesen. Die Daten werden als CSV-Strings pro Objekt bereitgestellt.

Ein Simulatorobjekt entfernen

(REMOVEOBJECT:objectid)

Um ein Objekt zu entfernen, benötigen Sie die ObjectID, die von CREATEOBJECT generiert wird (REMOVEOBJECT:@objid_1) entfernt die Giraffe, die wir zuvor erstellt haben

Eine Route an ein Simulatorobjekt senden

Einige SimObjekte können sich bewegen, wenn ihnen eine Wegpunktliste gesendet wird. Dazu gehören die meisten Bodenfahrzeuge sowie einige Tiere und Personen. Allerdings bewegen sich nicht alle Objekte, wenn man ihnen eine Route sendet. Beachten Sie außerdem die Kollisionserkennung des Simulators: Normalerweise bewegen sich Objekte nicht in die Nähe Ihres eigenen Fahrzeugs.

(SENDWAYPOINTS:objectid,filename,loop)

- *objectid* ist wiederum die ID, die von CREATEOBJECT generiert wurde
- *filename* ist der Name einer Datei mit der Wegpunktliste im Scripts-Ordner von AAO:
Dokumente\LorbyAxisAndOhs Files\Scripts\
- *loop* ist 0/1 und teilt dem Simulator mit, ob sich das Objekt ununterbrochen auf der Route bewegen soll.

Beispiel:

(SENDWAYPOINTS:(L:objid_1),relwps.csv,1)

wobei die Datei „relwps.csv“ wie folgt aussieht:

```
REL, 90, 400, 0, 5, 1  
REL, 180, 450, 0, 5, 1  
REL, 270, 400, 0, 5, 1  
REL, 0, 500, 0, 5, 1
```

Die Wegpunkte in der Datei haben folgende Syntax:

ABS|REL,x,y,z,Geschwindigkeit,Bodenweg

- *REL|ABS*: bezeichnet die Art der Positionierung eines Objekts, entweder RELativ zum letzten Wegpunkt oder an einer ABSoluten Breiten-/Längenposition
- *x,y,z*:
bei Verwendung von REL sind dies „Peilung, Entfernung, Höhenversatz“ relativ zum letzten Wegpunkt
bei Verwendung von ABS sind dies „Breitengrad, Längengrad, Höhe“
- *speed* ist die Geschwindigkeit in Knoten am Wegpunkt
- *onground* ist 0/1 und teilt der Simulation mit, ob sich der Wegpunkt in der Luft oder am Boden befindet

Fenster mit AAO erfassen und an anderer Stelle anzeigen

Mit dem folgenden Befehl können Sie jeden beliebigen Prozess oder jedes beliebige Unterfenster auf Ihrem Computer erfassen und in einem AAO-FIP-Gauge oder auf einer AAO-Webseite anzeigen. Auf diese Weise können Sie beispielsweise ein abgekoppeltes Simulator-Instrument auf Ihr Tablet oder Smartphone streamen.

(CAPTURETOFIP:windowname|lvarname|mode|parameter1|parameter2...)

- „windowname“ ist entweder der Prozessname oder der Fenstertitel
(CAPTURETOFIP:msegedge|edgefull|FULL|RESIZE~960~540|NOBLACK)
 - Wenn es mehrere Fenster mit demselben Titel gibt, können Sie ein bestimmtes Fenster durch einen Index nach dem Zeichen „~“ angeben:
(CAPTURETOFIP:WASMINSTRUMENT~0|fs0full|FULL|RESIZE~512~512|MSXY~1.0~1.0|REPOS)
(CAPTURETOFIP:WASMINSTRUMENT~1|fs1full|FULL|RESIZE~512~512|REPOS~10000~10000)
 - Es ist möglich, nur einen Teil des Fenstertitels, beginnend am Anfang, zu verwenden, indem man „~PART“ an den Fensternamen anhängt.
- Um die Fenstertitel zu finden, verwende „Tools->Show all current windows“
- „varname“ ist der Name einer String-LVar deiner Wahl. Nach dem Erfassen des Fensterinhalts speichert AAO die Bilddaten als Base64-kodierte Zeichenkette in dieser Variablen (PNG-Format)

- Mit „mode“ können Sie das gesamte Fenster oder nur einen Ausschnitt davon erfassen
 - „FULL~Breite~Höhe[~Bits]“ erfasst das gesamte Fenster und passt dessen Größe auf Breite und Höhe an
 - „CROP ~OffsetX~OffsetY~Breite~Höhe[~Bits]“ schneidet einen Ausschnitt mit den Maßen Breite × Höhe aus dem Fenster aus, beginnend bei OffsetX und OffsetY
 - [optional:~bits] Fügt die Bittiefe als „~32“ oder „~24“ hinzu (Standard ist 32 Bit)

- Parameter: Sie können die folgenden optionalen Parameter hinzufügen
 - „RESIZE~Breite~Höhe“: Größe des erfassten Bildes anpassen
 - „REPOS~x~y“: Dies verschiebt das Quellfenster an die Position x,y. Dies ist beispielsweise nützlich für nicht angedockte Instrumente, um das Original aus dem Bildschirmbereich zu verschieben „REPOS~10000~10000“
 - „NOBLACK“: Bei manchen Aufnahmen kommt es gelegentlich zu schwarzen Bildflackern. Mit diesem Parameter können Sie AAO anweisen, alle rein schwarzen Bilder zu ignorieren.

Zur Maussteuerung:

Die Maus funktioniert nicht immer. Sie können die Maus nur in den eigentlichen App-Fenstern verwenden. Sie funktioniert nicht bei ausgeblendeten Fenstern, Pop-ups, Unterbildschirmen oder Elementen, die aus dem Bildschirmbereich verschoben wurden.

- „MSXY~scaleX~scaleY“: Erfassen und Skalieren des linken Mausklicks und des Mauseisens: ScaleX und ScaleY sind standardmäßig auf 1,0 eingestellt. Sie können diesen Wert nach oben oder unten anpassen, je nachdem, wo die Mausklicks auf dem Originalfenster landen sollen.

- Dadurch wird das Hauptfenster aktiviert, der Mausklick übertragen und anschließend an Ihre aktuelle Position zurückgesprungen.
„MSSIMPLE“: Sendet das Drücken bzw. Loslassen der linken Maustaste als interne Windows-Meldung, ohne das Zielfenster zu aktivieren.

Beispiel: AAO-Gauge, das Ihren Webbrowser anzeigt, auf 960 x 540 Pixel skaliert, mit 10 FPS und Maussteuerung

```
<?xml version="1.0" encoding="utf-8"?>
<Gauge Name="Edge Capture Full" Version="1.0">
<Position X="0" Y="0"/>
<Image Name="(L:edgefull, String)" ImageSizes="960,540,960,540"/>
<Update>
<Frequency>10</Frequency>
<Script>
(CAPTURETOFIP:msedge|edgefull|FULL|RESIZE~960~540|MSXY~1.0~1.0|NOBLACK)
</Script>
</Update>
<MouseVariable>edgefull_mouse</MouseVariable>
</Gauge>
```

Zusätzliche Befehle:

- (CAPTUREWINDOWRESET): Alle CAPTURETOFIP-Vorgänge zurücksetzen und neu starten
- (CAPTUREWINDOWCLOSE): Alle erfassten Fenster schließen

6 Verwendung anderer Skriptsprachen als RPN

Anstelle von RPN-Code können Sie Skripte in allen Sprachen schreiben, die der Windows Script Host (WSH) unterstützt, sowie in der gängigen Sprache JavaScript.

Um diese Funktionen nutzen zu können, fügen Sie Ihrem Skript den Skript-Header „(WSH:Sprache“ vor. Ihre Skriptsprache muss in der Lage sein, Objekte unter Verwendung der Syntax „obj.property“ zu verarbeiten.

Beispiel für Jscript:

```
(WSH:jscript|AaoEntry)
function AaoEntry() {
    for(var i = 0; i < 1000; i++){
        (L:mylvar) = 1+7+(L:mylvar);
    }
    (L:mylvar, ·String) ·= ·GetTitle();
    for(var i = 0; i < 10; i++){
        (NUMARR:testarr).push(i + (NUMARR:testarr).count());
    }
    (NUMARR:testarr).set(5, (NUMARR:testarr).get(5) + 1999);
    AaoCmd.exec("(SPEAK:This is a Test!)");
    return;
}

function GetTitle() {
    return ·(A:TITLE, ·String) ·+ ·" ·-> ·tested";
}
```

```
}
```

Beachten Sie, dass JScript nicht mit JavaScript identisch ist. Beide Sprachen haben zwar die gleiche Syntax, sind aber nicht dasselbe. In JScript gibt es weder „let“ noch „const“, es gibt keine Objekte, Klassen, Importe oder Lambda-Ausdrücke. In WHS fehlt zudem der gesamte Webkontext, sodass alles, was von „window“ oder „document“ abhängt, nicht funktioniert.

Beispiel für VBScript

```
(WSH:vbscript|AaoEntry)
Function AaoEntry()
    Dim i
    Dim aVal
    For i = 0 To 1000
        (L:mylvar) = 1+7+(L:mylvar)
    Next
    (L:mylvar, String) = MyStringFunc((L:mylvar, String))
    For i = 0 To 10
        (NUMARR:testarr).push(i + (NUMARR:testarr).count())
    Next
    aVal = (NUMARR:testarr).get(5)
    (NUMARR:testarr).set 5, aVal + 1999
    AaoCmd.exec("(SPEAK:This is a Test!)")
End Function

Function MyStringFunc(lvar)
    MyStringFunc = lvar + ", seven"
End Function
```

Webressourcen (englisch)

[Windows Script Host – Wikipedia](#)

[JScript-Sprachreferenz \(Windows Scripting – JScript\) \(archive.org\)](#)

Syntaxregeln für WSH:

- Die Methode, die bei Aktivierung des Skripts aufgerufen werden soll, muss im WSH:-Header angegeben werden: (WSH:Sprache|Methode).
- Die Methode kann im asynchronen Modus aufgerufen werden (WSH:Sprache|Methode|ASYNC). Stellen Sie sicher, dass Sie einen Ausstiegspunkt für das Ende der Methode mit „AaoCmd.exit()“ angeben
- Parameter für diese Methode können durch ein Komma hinzugefügt werden: (WSH:Sprache|Methode,p1,p2,...)
- Beachten Sie die AAO-Skriptparameter: Verwenden Sie „param1“, „param2“ usw. nicht im Code, es sei denn, dies ist erforderlich (= beim Aufruf des Skripts als K-Event mit Parametern)
- Simulatorvariablen werden direkt in den Code geschrieben, mit derselben Syntax wie in RPN.
- Das Setter-Symbol „(>“ wird in WSH nicht verwendet, Variablen werden direkt mit „=“ zugewiesen
- Simulator-Ereignisse (K:) werden ebenfalls nicht mit der „(>“-Syntax geschrieben. Stattdessen rufen Sie die Funktion „exec“ für sie auf und übergeben den Wert als Parameter:
(K:HEADING_BUG_SET) .exec (240) ;

- Dynamische Listen (NUMARR, STRARR) können verwendet werden, sind jedoch keine nativen Skript-Objekte. Sie müssen dieselben Befehle wie in RPN anwenden (push, pop usw.; siehe das obige Beispiel und die Tabelle im Kapitel über „RPN-dynamische Listen“)
- Hashmaps (HASMMap) können ebenfalls verwendet werden, sind jedoch ebenfalls keine nativen Skript-Objekte. Sie müssen dieselben Befehle wie in RPN anwenden (mget, mset usw., genau wie bei den dynamischen Listen; siehe Tabelle im Kapitel über „RPN-HashMaps“ oben).

Spezielle AAO-Funktionen für WSH

- Das „AaoCmd“-Objekt:
 - Die speziellen AAO-Befehle oder anderer RPN-Code können mit „exec“ (asynchron) und „execwait“ (synchron, es sei denn, Sie rufen eine asynchrone Funktion auf!) aufgerufen werden:


```
AaoCmd.exec („(SPEAK:Dies ist ein Test!)“);
AaoCmd.execwait („(SPEAK:Dies ist ein Test!)“);
```
 - AaoCmd verfügt über „setTimeout/clearTimeout“ und „setInterval/clearInterval“. Sie funktionieren genauso wie in JavaScript (sind aber in allen WSH-Sprachen verfügbar)


```
var myIntv = AaoCmd.setInterval („AddUp“, 250);
...
AaoCmd.clearInterval (myIntv);
```

- `AaoCmd.sleep(nnnn)` ; hält den Thread für nnnn Millisekunden an
- `AaoCmd.exit()` ; beendet den Thread
- `AaoCmd.log(text)` ; schreibt den Text in das Protokoll unter `\Documents\LorbyAxisAndOhs Files\Scripts`
Anstelle von `.log` können Sie auch `.info` und `.error` verwenden
- Sie können Simulatorvariablen dynamisch mit `getnumval/setnumval` und `getstrval/setstrval` abrufen/setzen. Der Variablenname muss ohne die Klammern „(– „)“ angegeben werden!
Stellen Sie außerdem sicher, dass die Variablen, die Sie auslesen möchten, initialisiert wurden, da sie sonst den Wert „0“ zurückgeben.
Dieses Beispiel liest den Status der Türen der PMDG 777 aus. Beim ersten Aufruf wird ein falscher Wert zurückgegeben, da die Variablen nicht initialisiert sind. Nach diesem ersten Aufruf geben alle nachfolgenden Aufrufe den korrekten Wert zurück.

```
(WSH:jscript|AaoEntry)
function AaoEntry() {
    (L:doorsclosed) = 1;
    for( var i = 0; i < 16; i++){
        var vStr = 'PMDG:DOOR_state[' + i + ']';
        if (AaoCmd.getnumval(vStr) == 0) {
            (L:doorsclosed) = 0;
            return;
        }
    }
}
```

- Es ist auch möglich, das Ergebnis von RPN-Berechnungen mit `AaoCmd.rpnnumval(code)`, `AaoCmd.rpnstrval(code)` abzufragen

```
if (AaoCmd.rpnnumval(,test.txt` fileexist) == 1) {  
    (L:fileexists) = 1;  
    return;  
}
```

- Einbinden:

- Sie können andere Skripte mithilfe von `<include>`-Tags in Ihren Code einbinden:

```
(WSH:jscript|AaoEntry)  
<include src="Tester-Alibrary" lang="jscript" alias="testlib" >  
function AaoEntry(){  
    (...)
```

In diesem Fall binden wir ein weiteres JScript namens „Alibrary“ ein, das sich in der Gruppe „Tester“ befindet:

```
(WSH:jscript|speak)  
function speak(vstr){ AaoCmd.exec(„(SPEAK:Here ` + vstr + `)“);}  
function add(n1,n2){ return n1 + n2;}
```

- Wir können dann den definierten Alias verwenden, um die Methoden dieser Bibliothek in unserem Hauptskript aufzurufen
 - `testlib.call(„speak“, „tester“);`
Ruft die Methode „speak“ aus der Bibliothek auf und übergibt „tester“ als Parameter
 - `var test = testlib.call(„add“, 5, 3);`
Ruft die Methode „add“ aus der Bibliothek auf und gibt „8“ zurück.

- Das Skript, auf das im <include>-Tag verwiesen wird, kann in jeder gültigen WSH-Sprache geschrieben sein. Sie können beispielsweise eine VBScript-Bibliothek erstellen, die Sie dann aus Ihrem JScript aufrufen.

- **Einfügen:**

- Es ist auch möglich, den Code aus einem anderen Skript in das aktuelle einzufügen:

```
<insert src="Tester-Alibrary">
```

- Dies ist ein einfacher Textersatz. AAO fügt den Code aus dem Quellskript an genau der Position des <insert>-Tags in Ihr aktuelles Skript ein. Das eingefügte Skript muss dieselbe Sprache verwenden.

Sowohl <include> als auch <insert> können entweder auf Skripte aus der AAO-Datenbank (src=„group-name“) oder auf Skriptdateien verweisen, die sich in \Documents\Lorby AxisAndOhs Files\Scripts befinden (src=„filename“).

JavaScript

Zusätzlich zu den oben genannten Möglichkeiten können Sie auch regulären JavaScript-Code verwenden. Der Code wird in einer Chromium-Umgebung ausgeführt. Alle Standard-Sprachfunktionen stehen zur Verfügung.

Wenn Sie in JavaScript asynchrone Methoden verwenden, wie beispielsweise Ereignisbehandler, stellen Sie sicher, dass das Skript im ASYNC-Modus ausgeführt wird, und geben Sie einen Ausstiegspunkt mit „AaoCmd.exit();“ an, wenn es beendet wird.

```
(WSH:javascript|AaoEntry|ASYNC)
var socket;
var socketOpen = false;
var myLoop;
function AaoEntry() {
  socket = new WebSocket('ws://127.0.0.1:41716');
  socket.addEventListener('message', (event) => {
    var message = event.data;
  });
  socket.addEventListener('open', (event) => {
    socketOpen = true;
  });
  socket.addEventListener('error', (event) => {
    socketOpen = false;
  });
  myLoop = setInterval(EndLoop, 250);
}

function EndLoop() {
  var endCheck = (L:endmyloop);
  if (endCheck == 1) {
    clearInterval(myLoop);
    AaoCmd.exit();
  }
}
```

Hinweise zur Performance

Von allen Skriptoptionen

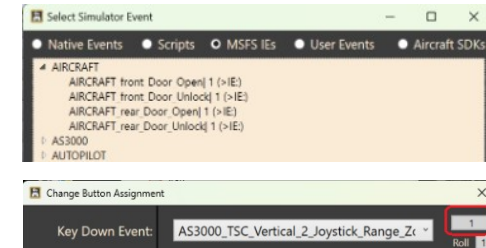
- ist RPN die schnellste und leistungsintensivste.
- sind WSH-Skripte bei der Ausführung etwas langsamer.
- ist JavaScript am wenigsten effizient und verursacht einen enormen Overhead. JavaScript sollte nur für Vorgänge verwendet werden, die auf andere Weise nicht realisierbar sind.

Achten Sie bei WSH und JavaScript darauf, dass das Skript ordnungsgemäß beendet wird, insbesondere im ASYNC-Modus.

7 MSFS-Eingabeereignisse

Eingabeereignisse sind flugzeugspezifisch, daher werden sie bei jedem Laden eines neuen Flugzeugs gelesen. Flugzeuge müssen in AAO keine Eingabeereignisse verwenden, daher kann es vorkommen, dass AAO keine sammelt.

Sie finden die Eingabeereignisse in der Baumliste des Ereignisauswahldialogs auf der Registerkarte „MSFS Ies“.



Eingabeereignisse benötigen einen korrekten Wert, um zu funktionieren (z. B. -1, 0, 1, 2)

Eingabeereignisse im RPN-Code

Zur Verarbeitung von Eingabeereignissen wurde der Header „IE:“ eingeführt. Wie bei K:-Ereignissen benötigen Eingabeereignisse keine Einheit. Sie können in Eingabeereignisse mit(>IE: schreiben und aus ihnen mit (IE: lesen.

Verkleinern der G3000-Karte: 1 · (>IE:AS3000_TSC_Vertical_2_Joystick_Range_Zoom)

Vergrößern der G3000-Karte: 0 · (>IE:AS3000_TSC_Vertical_2_Joystick_Range_Zoom)

Eingabeereignisse können mehr als einen Wertparameter haben!

Wenn Sie auf IEs stoßen, die auf den ersten Blick nicht zu funktionieren scheinen, versuchen Sie, sie mit mehr als einem Wert aufzurufen:

-1 1 (>IE:2:AS1000_RANGE_MFD)

(dies bewegt den Range-Joystick auf dem MSFS2024 G1000 MFD in X- und Y-Richtung)

8 Das AxisAndOhs In-Game-Panel

AxisAndOhs installiert ein eigenes In-Game-Panel in MSFS. Dieses Panel greift auf die WebAPI zu. Bitte lesen Sie unbedingt das Kapitel über die API in diesem Handbuch und informieren Sie sich über die erforderliche Konfiguration. Wenn das Panel keine Verbindung zur API herstellen kann, wird das Panel-Fenster rot umrandet.



1. Öffnen oder schließen Sie andere Ingame-Panels mithilfe von RPN-Skripten.
2. Verarbeiten Sie Coherent.Call- und .Trigger-Befehle, die Sie ausführen möchten.
3. Zeigen Sie AxisAndOhs-Web-FIPs im Panel-Fenster an.
4. Zeigen Sie andere Webseiten oder Assets an.

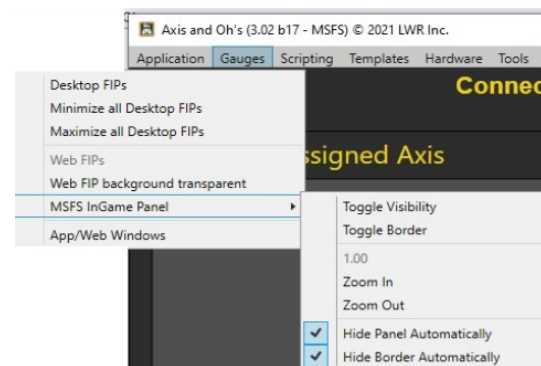
Allgemeine Bedienung

Das Panel muss lediglich aktiv sein – klicken Sie auf das Symbol in der MSFS-Menüleiste, sodass der Rahmen weiß wird.



Das Panel muss nicht geöffnet sein. Wenn Sie es nicht zur Anzeige von FIPs oder Webseiten verwenden, können Sie das Panel ausblenden und gleichzeitig aktiv lassen, indem Sie Folgendes tun:

- entweder die Standard-Schaltfläche „x“ verwenden
- oder das AAO-Menü „Gauges->MSFS InGame Panel“ verwenden



- oder verwenden Sie RPN-Skripte, um
 - (L:AAO_PANEL_HIDDEN, Zahl) auf 1 für „ausgeblendet“ und auf 0 für „sichtbar“ zu setzen
 - (L:AAO_PANEL_BORDER_HIDDEN, Zahl) auf 1/0 setzen, um den Rahmen des Fensters ein- bzw. auszublenden
 - LVar (L:AAO_PANEL_ZOOM, Zahl) auf einen Wert zwischen 0,0 und 10,0 setzen, um den Zoomfaktor zu ändern (0,0 und 1,0 entsprechen „kein Zoom“)

Andere In-Game-Panels öffnen und schließen

Sie können andere In-Game-Panels (ATC, Wetter, GSX, GTN750 usw.) mit dem folgenden RPN-Skriptcode umschalten:

```
, PANEL_ATC ` · (>L:AAO_TOGGLE_PANEL, ·String)
```

Die Panel-ID am Anfang finden Sie in der HTML-Datei des Panels. Diese befinden sich in der Regel in den Add-on-Paketen im Pfad „/html_ui/InGamePanels“. Oder suchen Sie mit Notepad++ in allen *.html-Dateien im Ordner /Packages von MSFS nach „panel-id“.

Coherent.call / Coherent.trigger

Mit den Lvars AAO_COHERENT_CALL und AAO_COHERENT_TRIGGER können Sie einfache Coherent-Befehle in RPN-Skripten aufrufen. Die Parameter lauten „COMMAND|ASSET_ID|Parameter“

Ein Coherent-Befehl kann bis zu 5 Parameter haben (getrennt durch das |-Symbol)

Beispiele:

```
,TOOLBAR_BUTTON_TOGGLE|PANEL_ATC|true`·(>L:AAO_COHERENT_CALL,·String)  
,TOOLBAR_BUTTON_MINIMIZED|PANEL_ATC|true`·(>L:AAO_COHERENT_CALL,·String)
```

Anzeigen von AxisAndOhs-Web-FIPs

Wenn Sie die Web-FIPs in AAO aktivieren (Gauges -> Web FIPs) und Instrumente zum Layout hinzufügen, zeigt das Ingame-Panel diese Instrumente in der Simulation an.

Dieses Verhalten kann mit dem folgenden Skript erzwungen werden:

```
„http://localhost:43280/webfips“·(>L:AAO_PANEL_URL,·String)
```

Um einzelne FIPs anzusprechen, fügen Sie deren numerische Position zur URL hinzu (Positionen beginnen bei 0)

```
„http://localhost:43280/webfips/1“·(>L:AAO_PANEL_URL,·String)
```

Anzeigen anderer Webseiten im Panel

Sie können das Panel so konfigurieren, dass es andere Websites anzeigt, sofern diese in einem iFrame dargestellt werden können.

Beispiele:

LittleNavMap: ,http://127.0.0.1:8965/` (>L:AAO_PANEL_URL, String)

Sky4Sim: ,http://127.0.0.1:18086/` (>L:AAO_PANEL_URL, String)

AxisAndOhs Web FMC: ,http://localhost:43380/webapi/a320cdu/index.html` (>L:AAO_PANEL_URL, ·String)

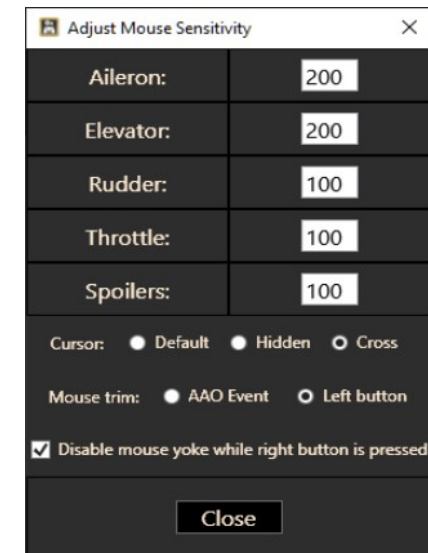
Fallback für die Verarbeitung von Bvars

Sollte der Standard-BVar-Hook innerhalb von AAO versagen, beginnt das InGame-Panel automatisch mit der Verarbeitung Ihrer BVar-Anfragen. Dazu muss das Panel in MSFS aktiv sein (weißer Rand in der Menüleiste).

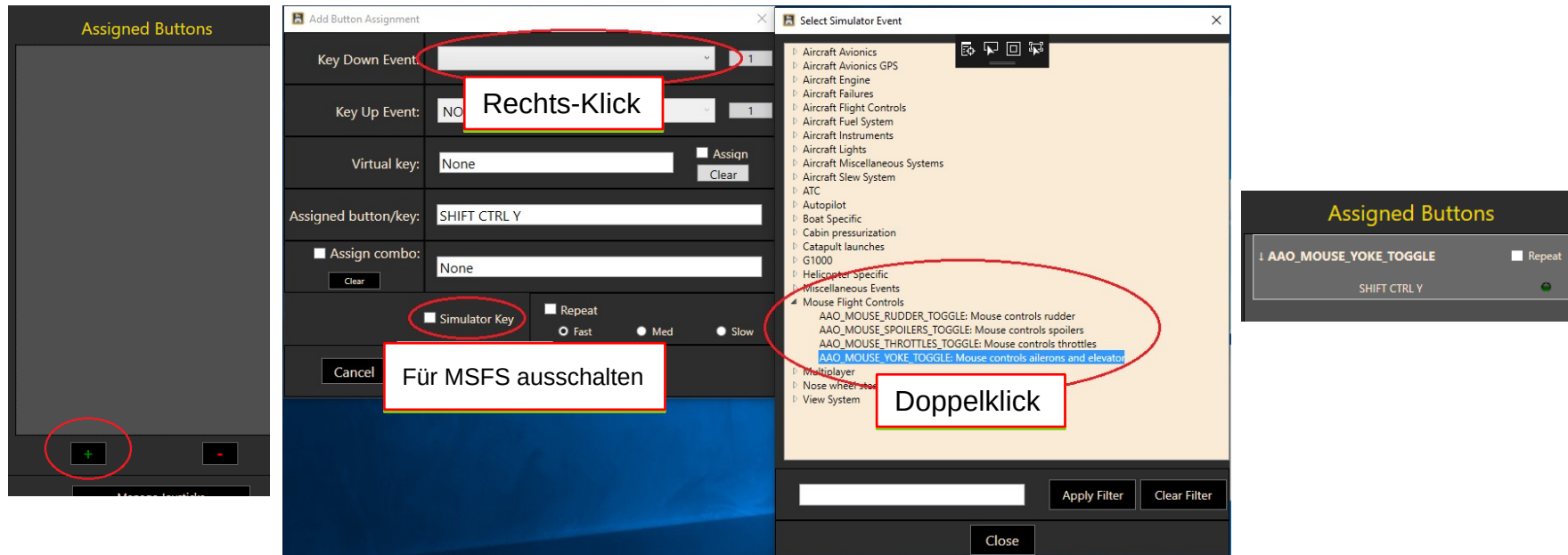
9 Maus als Steuerknüppel

AxisAndOhs bietet Optionen, um Ihre Maus als Flugsteuerungsgerät zu verwenden. Sie können die Maus als Steuerknüppel, Ruder, Gashebel oder Spoilerhebel nutzen. Die Mausempfindlichkeit und der Cursortyp können im Menü „Hardware“ geändert werden:

- Um die Empfindlichkeitswerte zu ändern, drehen Sie das Mousrad über den Zahlenfeldern
- Ein Wert von 100 entspricht einer 1:1-Übersetzung des Bildschirmpixelabstands in eine Achsenbewegung. Durch Erhöhen des Werts wird die Geschwindigkeit der Mausbewegung erhöht, durch Verringern des Werts verlangsamt.
- Sie können einen Cursortyp auswählen, der angezeigt werden soll, wenn das Maus-Steuerknüppel aktiv ist. Beachten Sie, dass dies eine systemweite Einstellung ist. Sollte AAO abstürzen, bleibt der Cursor so, bis Sie die App neu starten und den Maus-Steuerknüppel erneut aktivieren
- „Mouse Trim“: Zentriert den Mauszeiger neu, sodass du nicht mit der Maus „springen“ musst. Du kannst auswählen, ob du das Trimmen mit der linken Maustaste oder über ein AAO-Ereignis auslösen möchtest, das du einer Joystick-Taste oder einer Tastaturtaste zuweist
- Du kannst eine Option aktivieren, die das Maus-Yoke deaktiviert, solange du die rechte Maustaste gedrückt hältst.



Um auf die Maus-Yoke-Modi zugreifen zu können, müssen Sie die entsprechenden Umschaltbefehle Tasten oder Tastenkombinationen zuweisen:



In diesem Beispiel wurde „Umschalt-Strg-Y“ als Yoke-Umschalter festgelegt. Durch Drücken dieser Tasten wird der Maus-Yoke-Modus aktiviert, durch erneutes Drücken wird er deaktiviert.

Hinweis: Auf Windows-Systemen empfängt nur die App, die den Fokus hat, Tastaturereignisse. Wenn Sie Tastenkombinationen verwenden möchten, während das Maus-Yoke aktiv ist, müssen Sie die Mauszeiger-Einstellung auf „Standard“ belassen.

10 Erweiterte Energieverwaltung (Win 8.1 und höher)

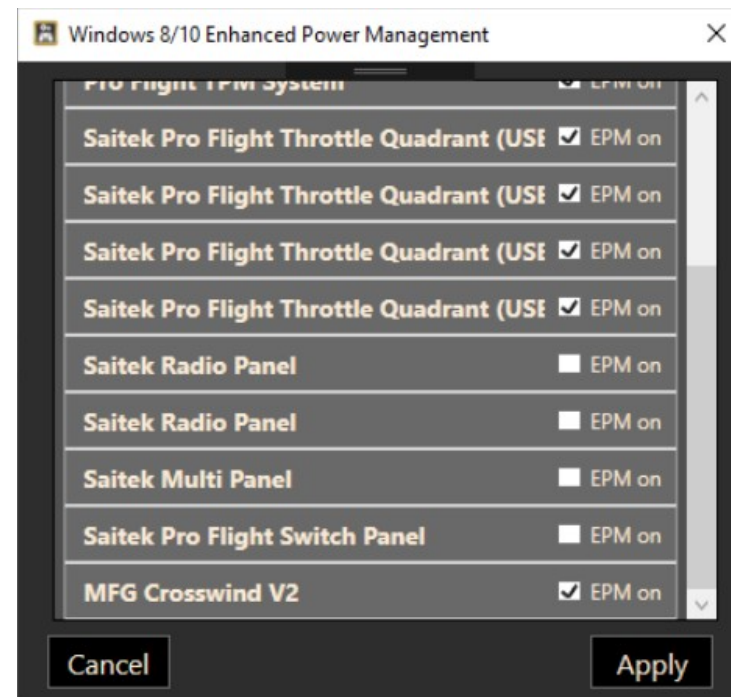
Ältere USB-Geräte weisen Probleme auf, wenn sie mit einem Computer verwendet werden, auf dem Windows 8 oder höher läuft, da diese Betriebssysteme über die Funktion „Erweiterte Energieverwaltung“ verfügen. Der auffälligste Fall ist das Saitek Multi Panel, bei dem das Display zwar aufleuchtet, aber keinen Text anzeigt.

Um dieses Problem zu beheben, können Sie das EPM über die Funktion „Win 8/10 Enhanced Power Management“ im Menü „Tools“ deaktivieren.

In diesem Dialogfeld sehen Sie alle von LAAO erkannten Geräte und deren aktuellen EPM-Status.

Deaktivieren Sie EPM, wenn Sie Probleme mit Geräten haben, wie z. B. fehlende Funktionen oder ständiges Verbinden und Trennen des Geräts.

Hinweis: Um dieses Dialogfeld nutzen zu können, muss die App „als Administrator“ ausgeführt werden.



11 Saitek-Panels

Lorby AxisAndOhs kann die Saitek-Geräte der Serien Radio-, Multi- und Switchpanel verwalten. Wenn Sie im Menü „Hardware“ die Option „Saitek aktivieren“ auswählen, sucht die App nach angeschlossenen Geräten und ordnet diese den entsprechenden Simulatorereignissen zu.

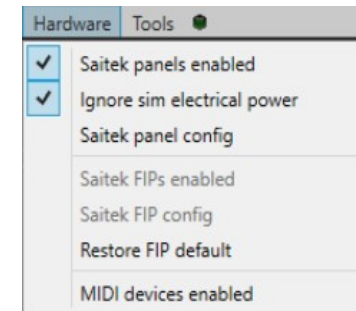
Sonderfunktionen des Radio-Panels:

- Durch eine Sekunde langes Gedrückthalten der Taste wird zwischen 25- und 8,333-Hz-Frequenzabstand umgeschaltet
- Im „DME“-Modus wechselt die Taste zwischen DME1 und DME2
- Die „ADF“-Einstellung zeigt links ADF1 und rechts ADF2 an. Halten Sie die Taste länger als 1 Sekunde gedrückt, um zwischen dem Drehregler für ADF1 und dem für ADF2 zu wechseln.
- In der XPDR-Einstellung zeigt das Gerät links den Transpondercode und rechts den aktuellen QNH an. Verwenden Sie die Taste, um zwischen der Bearbeitung des Transpondercodes und der Einstellung von Kohlsmann zu wechseln
 - Bearbeitung des Transpondercodes: Mit dem inneren Drehregler ändern Sie die Zahlen, mit dem äußeren Drehregler wählen Sie die zu ändernde Ziffer aus
 - Änderung von Kohlsmann: Mit dem inneren Drehregler ändern Sie den Wert, mit dem äußeren Drehregler wechseln Sie zwischen mb und inHg.

Konfiguration von Funk-, Schalter- und Multipanels

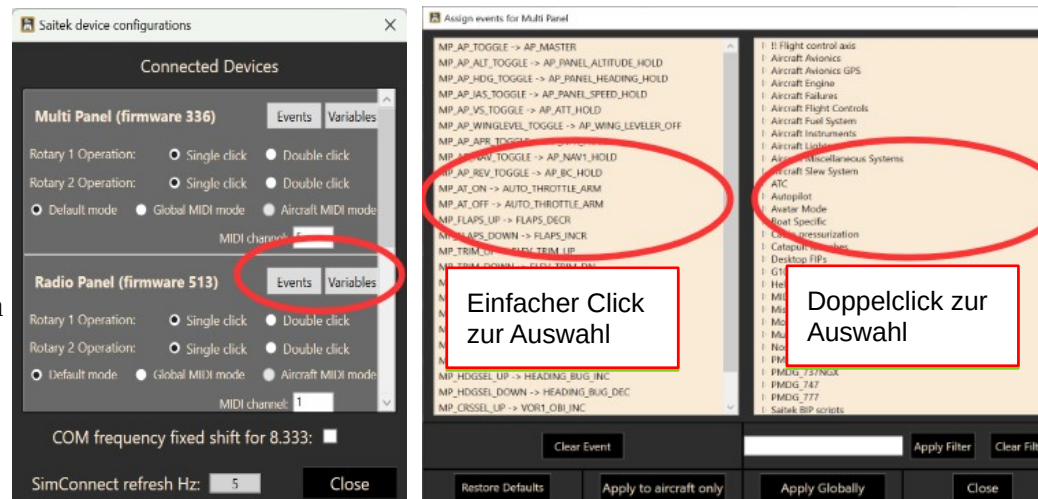
Im Menü „Hardware“ können Sie die Parameter für die Bedienung der Panels ändern

- „Saitek-Panels aktiviert“ schaltet die Panel-Verbindung ein oder aus
- „Sim-Stromversorgung ignorieren“ aktiviert die Panel-Anzeigen unabhängig davon, ob im aktuellen Flugzeug Strom verfügbar ist
- „Saitek-Panel-Konfiguration“ öffnet den Dialog zur Ereigniskonfiguration:



Die Drehgeber können auf Einzel- oder Doppelklick-Funktion eingestellt werden. Bei älteren Funk-Panels ist der Doppelklick die Standardeinstellung.

„MIDI-Modus“ deaktiviert alle Ereigniszuweisungen vollständig. Das Gerät sendet stattdessen MIDI-Ereignisse. Diese können Sie in AAO als Tasten zuweisen.



In den Dialogen „Events“ und „Variablen“ können Sie die Konfiguration eines Panels ändern, welche Ereignisse gesendet und welche Variablen angezeigt werden. Klicken Sie einmal auf den Eintrag, den Sie ändern möchten, in der linken Liste, und doppelklicken Sie dann auf das Ersatzereignis in der rechten Liste.

Das Verringern der SimConnect-Aktualisierungsrate (Mausrad) kann bei Rucklern im Simulator helfen (Standard: 18 Hz).

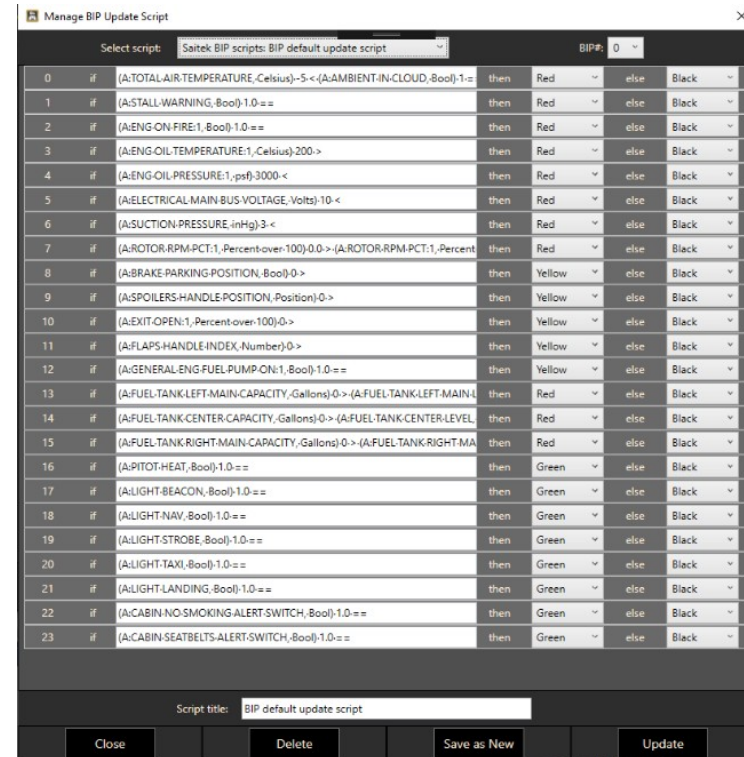
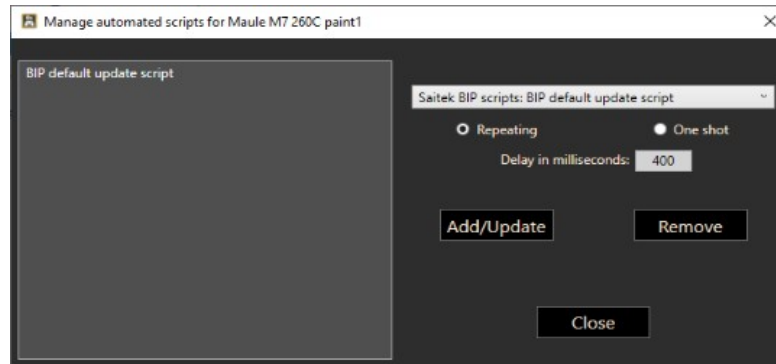
Hintergrundbeleuchtetes Informationspanel „BIP“

BIPs werden über RPN-Skripte unterstützt. Das Grundprinzip besteht darin, dass Sie ein automatisiertes Skript erstellen, das die LEDs am BIP entsprechend den Simulatorvariablen einstellt. AAO verfügt über einen speziellen Dialog zum Erstellen dieses Aktualisierungsskripts unter **„Hardware → Saitek BIP-Konfiguration“**.

Klicken Sie mit der rechten Maustaste in die großen Textfelder, um eine Simulatorvariablendefinition in das Skript zu kopieren.

Sie können jeden beliebigen RPN-Code eingeben, solange er ein boolesches Ergebnis von 0 oder 1 liefert, das die RPN-Klausel „if{“ verstehen kann.

Durch das Speichern oder Aktualisieren der Liste wird ein einzelnes RPN-Skript erstellt, das Sie Ihrem Flugzeug unter „Scripting → Automated scripts“ zuweisen können.



Es wird empfohlen, eine lange Verzögerung zu wählen, da die meisten Anzeigeleuchten nicht zeitkritisch sind.

Beachten Sie, dass jedes automatisierte Skript einer bestimmten Flugzeugkonfiguration zugeordnet ist!

Das Standard-Aktualisierungsskript bedient das folgende Layout:

#0 Vereisungsgefahr	#1 Strömungsabriss warnung	#2 Triebwerksbrand	#3 Öltemperatur zu hoch	#4 Öldruck zu niedrig	#5 Spannung zu niedrig	#6 Unterdruck zu niedrig	#7 Rotordrehzahl zu niedrig
#8 Feststellbremse	#9 Spoiler ausgefahren	#10 Tür offen	#11 Klappen ausgefahren	#12 Kraftstoffpumpe eingeschaltet	#13 Linker Tank fast leer	#14 Mittlerer Tank fast leer	#15 Rechter Tank fast leer
#16 Pitot-Heizung eingeschaltet	#17 Positionslicht	#18 Navigationslichter	#19 Blitzlicht	#20 Taxischeinwerfer	#21 Landelichter	#22 Rauchen verboten	#23 Anschnallhinweis

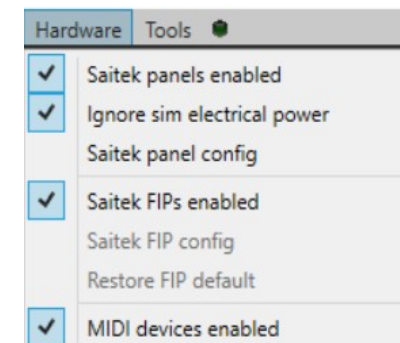
Flight Instrument Panels (FIP)

AxisAndOhs unterstützt die Integration von Saitek/Logitech „Flight Instrument Panel“-Geräten.

- Für das FIP muss der originale 64-Bit-Treiber von Logitech in der Version 8.0.150.0 installiert werden, der am 13.04.2018 veröffentlicht wurde. Dies ist die ältere Treiberversion, die noch über die von AAO benötigte API eines Drittanbieters verfügt.
- Es ist NICHT notwendig oder empfehlenswert, ein Plug-in von Logitech zu installieren!
- Vergewissern Sie sich jedes Mal vor dem Start von AxisAndOhs, dass die FIP-Geräte vom Treiber aktiviert wurden.
 - Die Tasten „Seite nach oben“/„Seite nach unten“ blinken einige Sekunden lang rot.
 - Der Bildschirm wechselt dann zur Standardanimation von Logitech:



Die Geräte können nur von AxisAndOhs verwendet werden, wenn sie sich in diesem Zustand befinden.



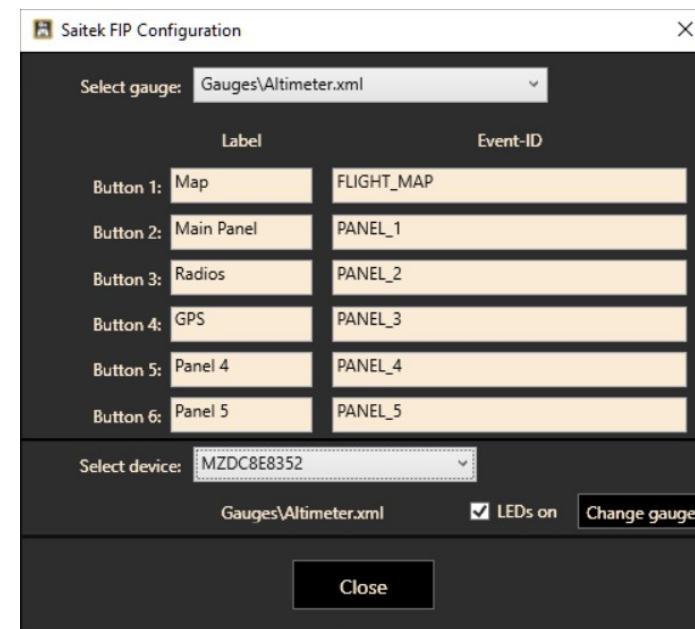
- Aktivieren Sie die FIP-Verbindung im Menü „Hardware“:

Wenn Sie Anzeigen ersetzen oder hinzufügen möchten, finden Sie die Definitionsdateien hier:
C:\Users\...\AppData\Local\LORBY_SI\LorbyAxisAndOhs\FipGauges_MSFS

FIP-Konfiguration

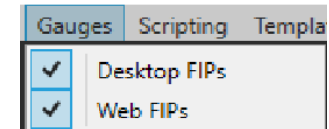
Wenn Sie im Menü „Hardware“ die Option „Saitek FIP config“ auswählen, öffnet sich der Konfigurationsdialog:

- Die Combobox „Select gauge“ enthält alle verfügbaren Anzeigen-Definitionen
- In der Spalte „Label“ können Sie den Text ändern, der neben den Tasten auf dem FIP angezeigt wird
- In der Spalte „Event-ID“ können Sie das Simulator-Ereignis ändern, das der Taste zugeordnet ist
Klicken Sie mit der rechten Maustaste auf den Text, um den Ereignisauswahldialog zu öffnen
- Die Combobox „Gerät auswählen“ enthält alle FIPs, die in der Konfigurationsdatei vorhanden sind
- Der Text unterhalb der Auswahl zeigt die Anzeige an, die dieses FIP anzeigt. Um sie zu ändern, wählen Sie oben eine Anzeige aus und klicken Sie dann auf „Anzeige ändern“
- „LEDs ein“ steuert, ob die Tasten auf dem FIP beleuchtet sein sollen oder nicht.



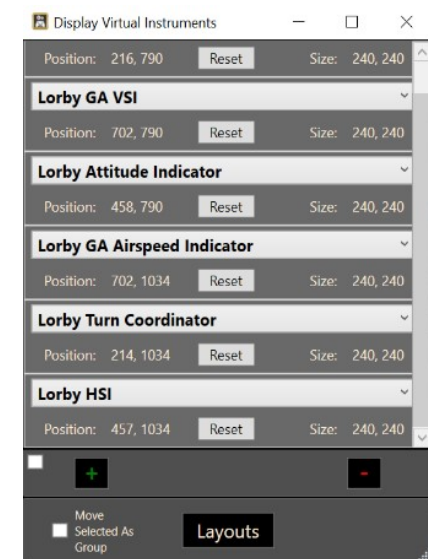
12 Desktop-FIPs

Mit der Funktion „Desktop-FIPs“ können Sie virtuelle Instrumente auf Ihrem Desktop erstellen. Um sie zu aktivieren, rufen Sie das Menü „Anzeigen“ auf und wählen Sie „Desktop-FIPs“



Dadurch wird die Verwaltungskonsole geöffnet:

- Um ein virtuelles Instrument hinzuzufügen, klicken Sie auf die Schaltfläche „+“.
- Um das angezeigte Instrument zu ändern, wählen Sie die gewünschte Anzeige aus der grauen Dropdown-Liste aus.
- Um ein Instrument zu entfernen, klicken Sie in das entsprechende Feld, sodass es rot wird, und klicken Sie dann auf die Schaltfläche „-“.
- Position und Größe können durch Drehen des Mauseis über den Zahlen geändert werden.
- Ausgewählte Anzeigen können über das Kontrollkästchen als Gruppe verschoben werden.



Die virtuellen Instrumente können an eine beliebige Stelle gezogen und in ihrer Größe angepasst werden. Wenn die Anzeige über Mausbereiche verfügt, lösen die vier Softkeys am unteren Rand die zugehörigen Ereignisse aus (Höhenmessereinstellung, Kurs, Flugrichtung usw.).

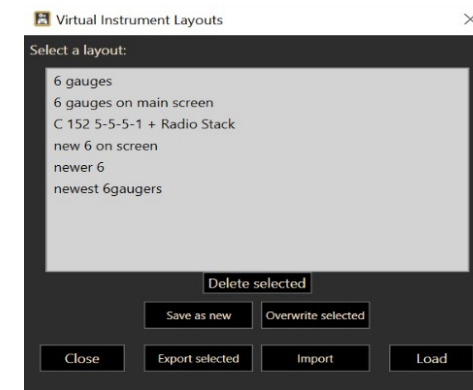
Speichern und Laden verschiedener Instrumentenlayouts

Sie können verschiedene Instrumentenkonfigurationen als „Layouts“ speichern.

Über die Schaltfläche „Layouts“ öffnen Sie den Verwaltungsdialog:

Sie können

- das aktuelle Instrumentenlayout als neuen Eintrag speichern oder einen bestehenden überschreiben
- ein zuvor gespeichertes Layout laden
- Layouts importieren und exportieren (Sie können für den Export mehrere Layouts auswählen)
- Layouts löschen



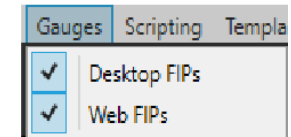
Spezielle Befehle für FIPs

AxisAndOhs verfügt über einige integrierte Spezialbefehle für die FIPs. Mit diesen können Sie FIPs öffnen, schließen, aktualisieren oder das Layout neu laden. Sie finden die Befehle in der Gruppe „Desktop-FIPs“ im Ereignisauswahldialog,

13 Web-FIPs

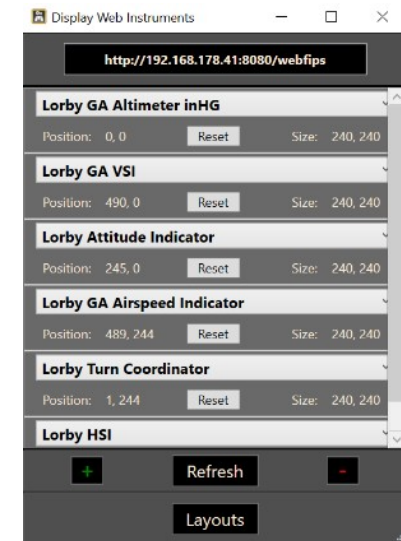
Für die Web-FIPs muss die AxisAndOhs-App „als Administrator“ ausgeführt werden!

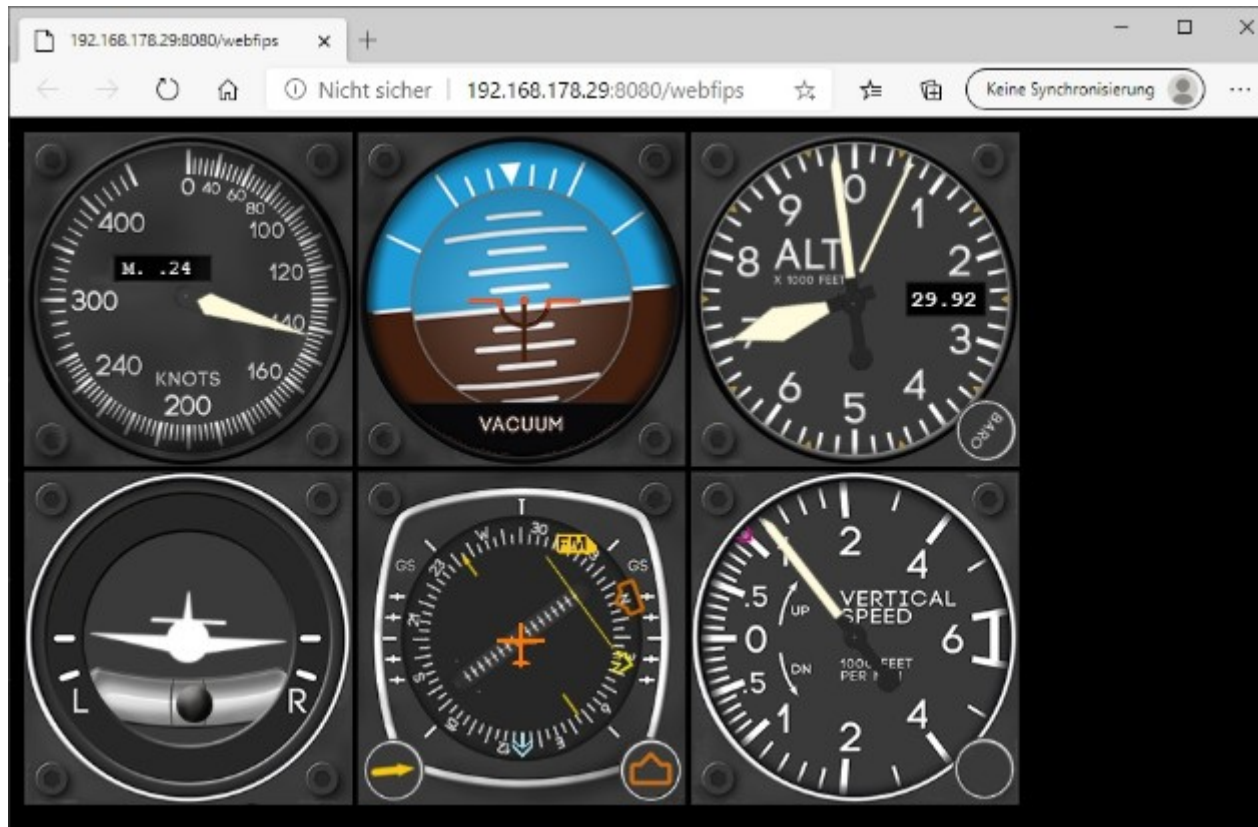
Mit der Funktion „Web-FIPs“ können Sie über einen standardmäßigen HTML5-fähigen Webbrowser auf einem Gerät, das über einen Netzwerkzugang zu Ihrem Computer verfügt, auf virtuelle Messgeräte zugreifen. Um diese Funktion zu aktivieren, rufen Sie das Menü „Gauges“ auf und wählen Sie „Web-FIPs“. Die Flugzeuglayouts funktionieren genauso wie bei den Desktop-FIPS, siehe vorherige Seite.



Dadurch wird die Verwaltungskonsole geöffnet:

- Das Textfeld oben zeigt die URL an, die Sie in die Adresszeile des Browsers eingeben müssen. Durch Doppelklicken darauf wird Ihr lokaler Standardbrowser mit dieser URL geöffnet
- Um ein Web-Instrument hinzuzufügen, klicken Sie auf die Schaltfläche „+“
- Um das angezeigte Instrument zu ändern, wählen Sie das gewünschte Instrument aus der grauen Dropdown-Liste aus
- Um ein Instrument zu entfernen, klicken Sie in das entsprechende Feld, sodass es rot wird, und klicken Sie dann auf die Schaltfläche „-“
- Position und Größe können durch Drehen des Mauseisens über den Zahlen geändert werden. Achten Sie darauf, auf „Aktualisieren“ zu klicken, um die Änderungen zu übernehmen





Wenn in Ihrem lokalen Netzwerk andere Anwendungen den Port 43280 nutzen, funktionieren die WebFIPs nicht. Sie können den Port über „Tools->Configure WebFIP port“ (Extras -> WebFIP-Port konfigurieren) ändern. Starten Sie die App nach der Änderung neu.

Zugriff auf die WebFIPs

Geben Sie die im Verwaltungsdialog angezeigte URL in Ihren Webbrowser ein:

`http://192.168.178.29:43280/webfips`

Auf die WebFIPs kann auch separat zugegriffen werden. Fügen Sie entweder den Index des FIPs, den FIP-Namen oder den Namen der XML-Datei an die URL an (die Indizes verlaufen von oben nach unten, beginnend mit 0)

`http://192.168.178.29:43280/webfips/0`

`http://192.168.178.29:43280/webfips/Lorby GA VSI`

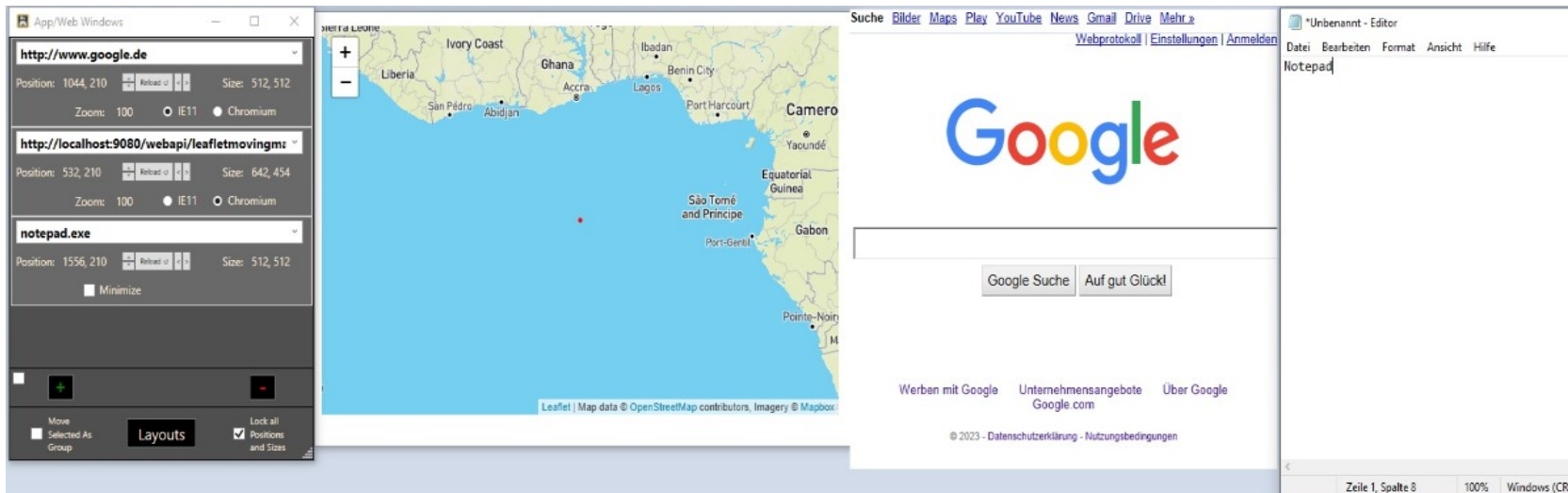
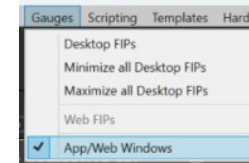
`http://192.168.178.29:43280/webfips/LorbyGaAltimeterHG.xml`

Spezielle Befehle für FIPs

AxisAndOhs verfügt über einige integrierte Spezialbefehle für die FIPs. Mit diesen können Sie FIPs öffnen, schließen, aktualisieren oder das Layout neu laden. Sie finden die Befehle in der Gruppe „Desktop-FIPs“ im Dialogfeld zur Ereignisauswahl, wenn Sie einer Schaltfläche eine Funktion zuweisen oder ein Ereignis zu einem Skript hinzufügen.

14 App- und Web-Fenster

Im Menü „Anzeigen“ finden Sie auch die Funktion „App-/Web-Fenster“.
Damit können Sie andere Apps starten und Webseiten an festen Positionen auf Ihrem Desktop anzeigen.



Der Verwaltungsdialog funktioniert genauso wie die Desktop- und Web-FIPs. Verwenden Sie „+“, um ein neues Fenster hinzuzufügen, und „-“, um es zu entfernen.

Geben Sie eine URL, den Pfad zu einer ausführbaren Datei (Befehlszeilenparameter können nach einem Komma angegeben werden) oder den Namen eines laufenden Prozesses ein, wie er im Task-Manager angezeigt wird, oder wählen Sie eine der AAO-Webseiten aus der Dropdown-Liste aus. Klicken Sie anschließend auf „Neu laden“.

Wenn die Zielanwendung bereits läuft, übernimmt AAO dieses Fenster, verschiebt es und passt seine Größe an.

Zunächst werden die Fenster als leere Anwendungen angezeigt, die Sie ziehen und in der Größe anpassen können. Wenn Sie sie mit der Option unten rechts im Dialogfeld „sperren“, werden die Titelleisten dieser Fenster ausgeblendet und die Zielanwendung in den Vordergrund geholt.

Position, Größe und Zoom können durch Drehen des Mousrads direkt über den Zahlen im Dialogfeld geändert werden. Auch hier handelt es sich um eine doppelte Aktion: Das Drehen des Rads über der linken Hälfte der Zahl bewirkt große Änderungen, über der rechten Hälfte kleine Änderungen.

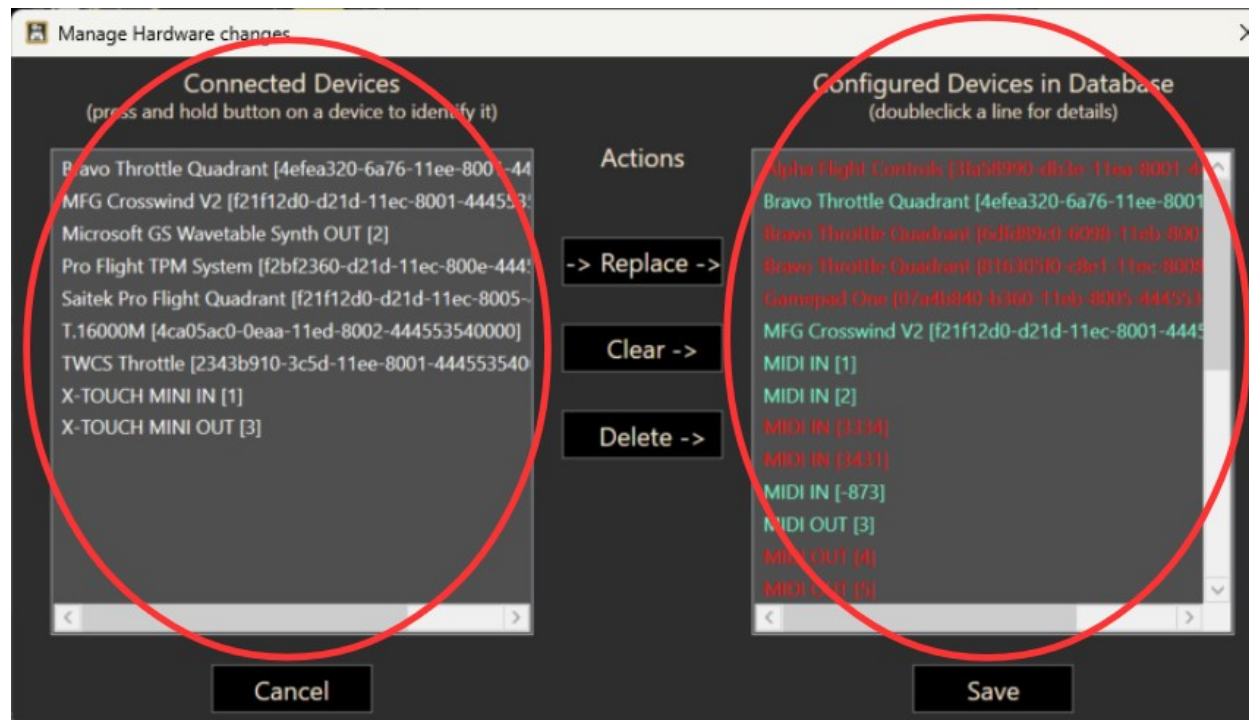
Verwenden Sie die Funktion „Layouts“, um Ihre Anordnung zu speichern. Wie bei den Desktop- und Web-FIPs wird ein geladenes Layout automatisch dem aktuell geladenen Flugzeug zugeordnet.

Webseiten können entweder im IE11- oder im Chrome-Kompatibilitätsmodus angezeigt werden. IE11 schont die PC-Ressourcen, Chrome bietet eine bessere Kompatibilität.

15 Hardware-Änderung

Wenn Sie Änderungen an Ihrer Controller-Konfiguration vornehmen, beispielsweise einen Joystick durch einen neuen ersetzen oder einen USB-Hub austauschen, müssten Sie theoretisch alle Ihre Steuerungen neu zuweisen. Um Ihnen dabei zu helfen, bietet Lorby AxisAndOhs einen Dialog, über den Sie Controller manuell neu zuweisen oder aus der AxisAndOhs-Datenbank löschen können. Rufen Sie diesen unter „Hardware → Hardware Change“ (Hardware → Hardware-Änderung) auf.

Die Liste auf der linken Seite zeigt alle Joystick-Geräte an, die derzeit an Ihren Computer angeschlossen sind



Die Liste auf der rechten Seite zeigt alle Joystick-Geräte an, die in der AxisAndOhs-Datenbank vorhanden sind.

Geräte, die mit einem angeschlossenen Joystick übereinstimmen, werden grün angezeigt, Geräte, die nicht mehr vorhanden sind, werden rot angezeigt

Doppelklicken Sie auf eine Zeile, um Details darüber zu erhalten, wo der Controller genau zugewiesen ist

-> **Replace** → (ersetzen)

Mit dieser Schaltfläche können Sie ein Joystick-Gerät in der Datenbank durch ein anderes (angeschlossenes) ersetzen. Ein typischer Anwendungsfall wäre, wenn Sie einen Joystick ausgetauscht haben oder wenn Windows die USB-Geräte, die die Joysticks darstellen, neu angeordnet hat (und Ihre Zuweisungen nicht mehr zu dem Controller passen, für den sie erstellt wurden).

- Wählen Sie das neue Gerät in der linken Liste aus
- Wählen Sie das zu ersetzende Gerät in der rechten Liste aus
- Drücken Sie „-> Replace →“
- Nach einem Sicherheitsdialog ersetzt die App den alten Joystick durch den neuen.
- Drücken Sie „Speichern“, wenn Sie mit den Änderungen zufrieden sind.

Clear →

Mit dieser Schaltfläche können Sie ein bestimmtes Joystick-Gerät als „leeren“ Joystick festlegen. Dies ist nützlich, wenn Sie einen Austausch vornehmen müssen, das neue Gerät jedoch bereits in der Datenbank vorhanden ist, oder wenn Sie einen bestimmten Controller deaktivieren möchten.

- Wählen Sie das Gerät, das gelöscht werden soll, in der rechten Liste aus
- Tippen Sie auf „Löschen →“
- Nach einem Sicherheitsdialog ersetzt die App den alten Joystick durch ein leeres Gerät.
- Tippen Sie auf „Speichern“, wenn Sie mit den Änderungen zufrieden sind.

Delete ->

Dadurch wird der ausgewählte Eintrag vollständig aus der rechten Liste entfernt. Verwenden Sie diese Funktion, wenn Sie Zuordnungen in der Datenbank haben, die veraltet sind oder die Sie nicht mehr verwenden.

Wählen Sie das Gerät, das gelöscht werden soll, in der rechten Liste aus

Tippen Sie auf „Löschen →“

Nach einem Sicherheitsdialog löscht die App das ausgewählte Gerät aus der Datenbank.

Drücken Sie „Speichern“, wenn Sie mit den Änderungen zufrieden sind.

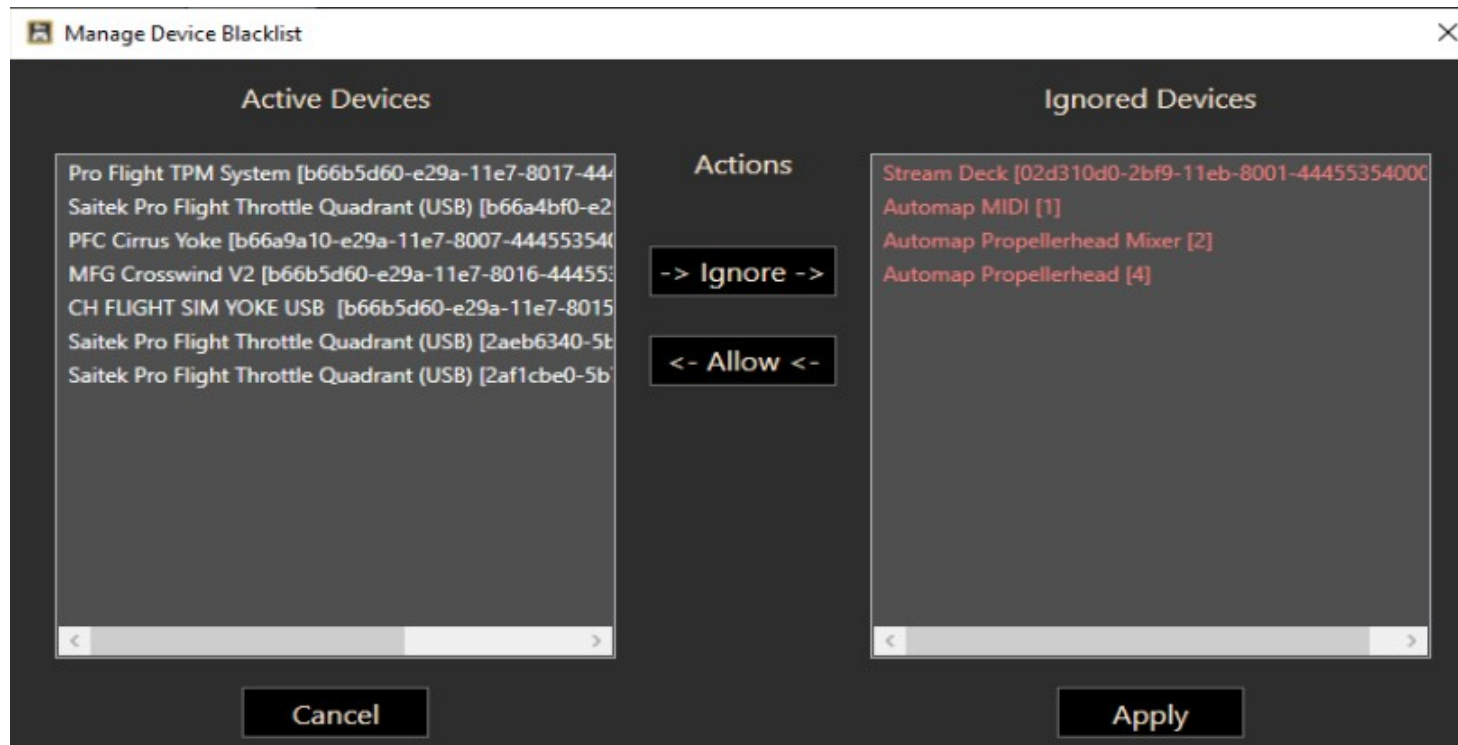
Joysticks trennen und wieder anschließen

Ein Gerät, das während der Ausführung der App getrennt wird, wird ignoriert. Wenn Sie es wieder anschließen, müssen Sie AxisAndOhs anweisen, nach Joystick-Geräten zu suchen, oder – noch besser – die App einfach neu starten.

Um nach Geräten zu suchen, können Sie die Option „Geräte erneut scannen“ im Menü „Hardware“ verwenden.

16 Geräte-Blacklist

Falls Sie andere USB-Geräte (keine Controller) besitzen, die AxisAndOhs mit unerwünschten Eingangssignalen überfluten, können Sie die App unter „**Hardware** → **Device Blacklist**“ so einstellen, dass diese Signale ignoriert werden.



17 Konfigurationsdateien, Datenbank und Protokolldateien

Datenbank

Lorby AxisAndOhs verwendet eine einfache XML-Datei, um alle Ihre Joystick-Zuweisungen zu speichern. Die Datei hat für jeden Simulator, auf dem Sie die App installiert haben, einen anderen Namen und befindet sich hier:

„C:\Users\...\AppData\Local\LORBY_SI\LorbyAxisAndOhsMSFS\ConfigDatabase.xml“

Die Datei wird in 4 „Generationen“ gespeichert. Sollte also etwas schiefgehen, können Sie zu einem früheren Zustand der Datei zurückkehren, indem Sie die n-te Sicherungskopie „ConfigDatabase_n.xml“ in „ConfigDatabase.xml“ umbenennen.

Startprotokoll

Diese Datei listet alle Ereignisse auf, die während des Starts der Anwendung aufgetreten sind. Sie dient zu Debugging-Zwecken, falls die App überhaupt nicht starten sollte.

„C:\Users\...\AppData\Local\LORBY_SI\LorbyAxisAndOhsMSFS\LAEO_Startup_log.txt“

Weitere Protokolldateien

Fehler in der App werden in Protokolldateien an diesem Speicherort aufgezeichnet. Wenn Sie doppelte Flugzeugdefinitionen im Simulator haben, beispielsweise nach einem Alpha-Update, werden diese ebenfalls hier in einer Protokolldatei aufgeführt.

App-Konfiguration

Einstellungen zur App selbst werden in dieser Datei gespeichert:

„C:\Users\...\AppData\Local\LORBY_SI\LorbyAxisAndOhsMSFS\LorbyAxisAndOhsConfiguration.xml“

Benutzerinhalte

Normalerweise werden alle Benutzerinhalte im Ordner „\Documents\LorbyAxisAndOhs Files“ gespeichert. Wenn dieser Ordner nicht verfügbar ist, beispielsweise auf OneDrive, wechselt AAO stattdessen zu

„C:\Users\...\AppData\Local\LORBY_SI\LorbyAxisAndOhsMSFS“.

Sie müssen dann Anzeigen usw. dort ablegen.

18 PMDG-Flugzeuge

AxisAndOhs stellt die Ereignisse des PMDG-SDK bereit, falls Sie PMDG-interne Funktionen Tasten oder MIDI-Controllern zuweisen möchten. Diese sind im Dialogfeld zur Ereigniszuweisung in separaten Gruppen aufgeführt.

Darüber hinaus bietet PMDG die Möglichkeit, Befehlscodes an das Ereignis (>K:ROTOR_BRAKE) zu senden. Diese Codes finden Sie in der Verhaltensdefinitionsdatei des jeweiligen PMDG-Flugzeugmodells.

Werterhöhungen

Eine gute Strategie für Werterhöhungen ist die Verwendung von RPN-Skripten anstelle einzelner Tastendrucke.

```
(L:PMDG_MCP_ALT) ·100 ·+ ·(>K:#84137) ·(L:PMDG_MCP_ALT) ·100 ·+ ·(>L:PMDG_MCP_ALT)
```

Dieses Skript verwendet eine lokale Variable, um die aktuelle AP-Höhe zu speichern, und erhöht diese bei einem Tastendruck um 100 Fuß – und sendet den Wert an die PMDG-Variable. Der Standardwert der PMDG-Variable ist 10000 ft, daher würde dieses Skript von einem automatisierten Skript profitieren, das einmalig beim Laden des Flugzeugs ausgeführt wird:

```
10000 (>L:PMDG_MCP_ALT)
```

Umschalten

Um die PMDG-Ereignisse wie Umschalter statt wie Ein-/Aus-Schalter funktionieren zu lassen, können Sie ebenfalls Skripte verwenden. Dieses Skript schaltet beispielsweise HDG_SEL in der PMDG 737 NGX um

```
1 (L:PMDG_MCP_HDG_SEL) - (>K:#70024) 1 (L:PMDG_MCP_HDG_SEL) - (>L:PMDG_MCP_HDG_SEL)
```

Mausereignisse

Es ist möglich, bestimmte Mauseaktionen an die Steuerelemente in einem PMDG-Cockpit zu senden. Die Mausereignisse befinden sich in der Gruppe „PMDG Mouse“ und dienen als Eingabeparameter für die >K-Ereignisse.

Ereignis	Zu sendender Wert
MOUSE_RIGHTSINGLE	2147483648
MOUSE_MIDDLESINGLE	1073741824
MOUSE_LEFTSINGLE	536870912
MOUSE_RIGHTDOUBLE	268435456
MOUSE_MIDDLEDDOUBLE	134217728
MOUSE_LEFTDOUBLE	67108864
MOUSE_RIGHTDRAG	33554432
MOUSE_MIDDLEDTRAG	16777216
MOUSE_LEFTDRAG	8388608
MOUSE_MOVE	4194304
MOUSE_DOWN_REPEAT	2097152
MOUSE_RIGHTRELEASE	524288
MOUSE_MIDDLERELLEASE	262144
MOUSE_LEFTRELEASE	131072
MOUSE_WHEEL_FLIP	65536
MOUSE_WHEEL_SKIP	32768
MOUSE_WHEEL_UP	16384
MOUSE_WHEEL_DOWN	8192

Beispiele:

```
16384 (>K:#69997)
8192 (>K:#69997)
```

Diese Skripte senden die Befehle „Mouse_Wheel_Up“ (16384) und „Mouse_Wheel_Down“ (8192) an das EVT_EFIS_CPT_BARO (#69997) (PMDG 737)

PMDG-Variablen auslesen

AAO kann auf die im PMDG SDK bereitgestellten Datenstrukturen zugreifen, sodass Sie diese Werte auch aus dem Simulator auslesen können.

Damit dies funktioniert, müssen Sie die Datenübertragung vom PMDG-Flugzeug aktivieren. Suchen Sie die Datei 7xx_Options.ini im Verzeichnis <MSFS_Directory>\LocalState\packages\pmdg-aircraft-7xx\work und stellen Sie sicher, dass Folgendes in dieser Datei steht:

```
[SDK]  
EnableDataBroadcast=1
```

Achten Sie darauf, mit einer leeren Zeile abzuschließen.

Diese Änderung erfordert einen Neustart des Simulators.

Nach der Aktivierung der Datenübertragung müssen Sie dem Flugzeug in AAO das entsprechende „Aircraft Automated“-Skript zuweisen. Sie finden die Skripte in der Gruppe „PMDG“; es gibt eines für jeden PMDG-Flugzeugtyp. Sie müssen mit „Repeating“ und einer angemessenen Aktualisierungsrate (200 ms oder schneller)

Wenn all dies eingerichtet ist, können Sie die Variablen wie folgt aus dem Flugzeug auslesen:

```
(PMDG:IRS_DisplaySelector)  
(PMDG:LTS_LandingLtRetractableSw[0])
```

Alle verfügbaren Variablen finden Sie im Dialogfeld „Variable einfügen“ des RPN-Skripteditors in den dafür vorgesehenen Gruppen.

Achten Sie auf Variablen vom Typ Array, wie die oben genannte. Wenn sie als „Array[]“ gekennzeichnet sind, kann hier auf mehr als einen Wert zugegriffen werden (0 – x).

Hinweis: Sie können diese Variablen nicht beschreiben. Verwenden Sie dafür die oben beschriebenen PMDG-Ereignisse.

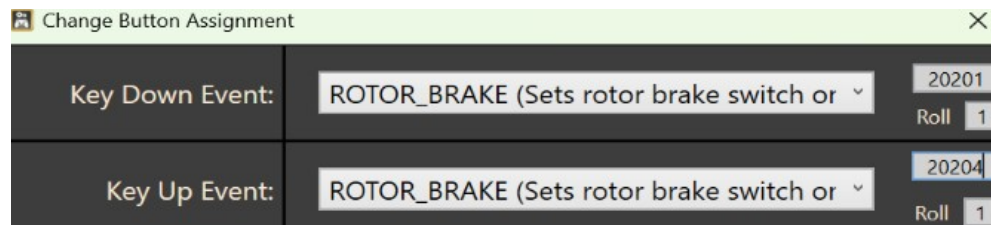
ROTOR_BRAKE-Codes

In MSFS besteht eine Alternative zur Verwendung der PMDG-Ereignisse darin, bestimmte Codes an das Ereignis (>K:ROTOR_BRAKE) zu senden. Die Codes lassen sich aus dem Verhaltenscode berechnen, der entweder in der entsprechenden Datei zu finden ist, wie z. B. `\Community\pmdg-aircraft-77w\SimObjects\Airplanes\PMDG 777-300ER\Behaviors\PMDG773ER_VC.xml` oder über den MSFS-Entwicklermodus unter „Tools->Behaviors“.

Für jedes Steuerelement wird der ROTOR_BRAKE-Code aus der Schaltnummer und der Mausektion berechnet, siehe Abschnitt „<CALLBACKCODE>“.

Beispiel:

Der FD-Schalter hat die Nummer „202“. Der Code für das Drücken und Loslassen der Taste berechnet sich wie folgt:
 $202 * 100 = 20200$ -> LeftSingle = 20201 -> LeftRelease = 20204



19 MIDI-Ausgang

Viele MIDI-Controller verfügen über LEDs, die beim Drücken einer Taste eine Rückmeldung geben. Diese LEDs können auch dazu verwendet werden, den Status einer Variablen im Simulator anzuzeigen. Dazu warten MIDI-Controller in der Regel auf MIDI-Out-Befehle, die auf einem bestimmten Kanal mit einer bestimmten Note und einer bestimmten Anschlagstärke ankommen.

Beispielsweise lauscht der Behringer X-Touch Mini auf Kanal 1, wobei die Note die Tastenposition angibt (0 ist die linke Taste der oberen Reihe, 15 ist die rechte Taste der unteren Reihe), Anschlagstärke 1 schaltet die LED hinter der Taste ein, 0 schaltet sie aus.

Ein Novation Launchpad erwartet einen Notenwert, der als „(16 * Reihe) + Spalte“ berechnet wird. Die Anschlagstärke ist binär codiert und recht kompliziert, aber beispielsweise steht 12 für „aus“ und 62 für „gelb mit voller Intensität“. Weitere Details findest du online im Launchpad-Programmers-Guide.pdf.

Um MIDI-Out-Events mit AxisAndOhs zu senden, musst du sie in ein RPN-Skript mit folgender Syntax einbetten:

velocity (>MIDI:<DeviceId>:<Action>:<channel>:<note>)

<Action> kann einer der folgenden Werte sein: „NoteOn“, „NoteOff“, „CC“, „PrgChng“

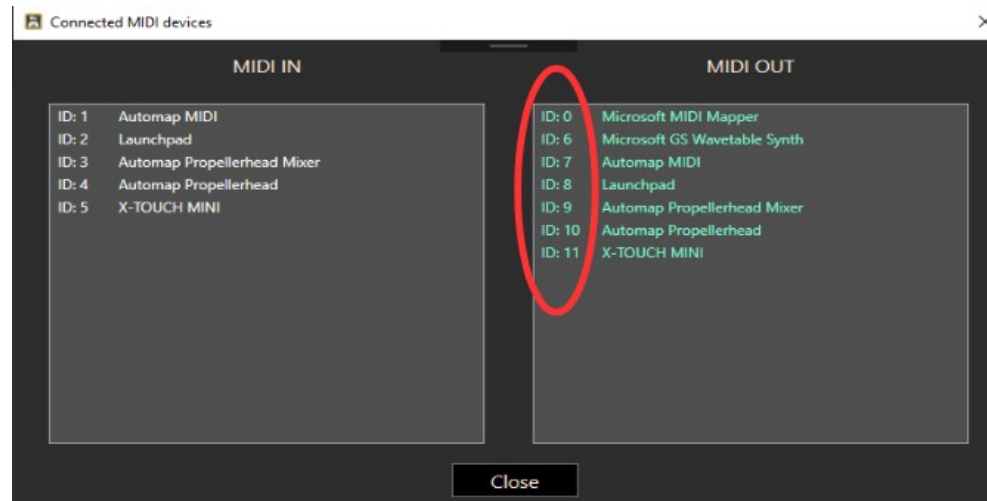
Beispiel:

„62 (>MIDI:8:NoteOn:1:17)“ färbt die zweite Taste in der zweiten Reihe eines Launchpads gelb.

„Program Change“ hat kein <note>: 1 (>MIDI:8:PrgChng:1)

Du kannst auch SysEx-Befehle senden: (**MIDIOUT:<DeviceId>:SysEx:[Zeichenkette aus 2-stelligen Hex-Werten]**)
(**MIDIOUT:8:SysEx:01020300AC**) führt zu F0 01 02 03 AC F7. Verwende in der Nachricht keine Werte über F0.

Die **DeviceId** findest du im Dialogfeld, das die verbundenen MIDI-Geräte anzeigt. (Hauptmenü → Hardware → MIDI-Geräte anzeigen)



Beispiel:

Schalte den Autopiloten mit der ersten Taste in der oberen Reihe des Behringer X-Touch um

Erstelle ein neues RPN-Skript „MIDI_SYNC_AP_ON“ (oder wie auch immer du es nennen möchtest) mit folgendem Inhalt:

```
(A:AUTOPILOT MASTER, Bool) 0 == if{ 0 (>MIDI:11:NoteOn:1:0) } els{ 1 (>MIDI:11:NoteOn:1:0) }
```

Weisen Sie es dann Ihrem Flugzeug als automatisiertes Skript zu, das alle 200 – 400 ms ausgeführt wird (es muss nicht öfter ausgeführt werden). Von nun an leuchtet die LED der ersten Taste am Behringer entsprechend auf, egal wie Sie den Autopiloten umschalten. Wenn Sie das AP_MASTER-Ereignis in AAO derselben Taste zuweisen, haben Sie eine AP-Umschalttaste mit visueller Rückmeldung.

20 Mehrere MIDI-Geräte – Kanalpriorität

Wenn Sie mehrere MIDI-Geräte verwenden, kann es vorkommen, dass sich die MIDI-IDs der Geräte ohne ersichtlichen Grund ändern. Dies liegt an der Art und Weise, wie Windows diese Geräte verwaltet: Die IDs werden bei jedem Neustart des Computers in der Reihenfolge vergeben, in der die MIDI-Geräte hochfahren. Infolgedessen funktionieren MIDI IN und OUT nicht richtig, da die IDs möglicherweise anders zugewiesen wurden als beim letzten Mal.

Um dieses Problem zu beheben, können Sie für jedes Gerät die „MIDI-Kanalpriorität“ aktivieren. Das bedeutet, dass das MIDI-Interface nun den MIDI-Kanal anstelle der Geräte-IDs als Zuordnungsschlüssel verwendet. So ist die Verwendung von MIDI im Grunde vorgesehen: Als das Protokoll in den 1980er Jahren entwickelt wurde, war der Kanal das Unterscheidungsmerkmal zwischen den Geräten.

Im Kanalprioritätsmodus müssen Sie sicherstellen, dass alle Ihre MIDI-Geräte auf unterschiedlichen Kanälen senden und empfangen (es gibt insgesamt 16 MIDI-Kanäle). Dies lässt sich in der Regel über die Konfigurationssoftware des jeweiligen Geräts einstellen.



21 Web-API

AxisAndOhs bietet eine auf Webtechnologie basierende API, um mit jeder Anwendung, die HTTP-Anfragen senden und empfangen kann, auf den Simulator zuzugreifen. Die API ist im Grunde ein Miniatur-Webserver. Mithilfe einer einfachen JSON-Struktur oder spezieller Anfrageparameter können Sie Ereignisse auslösen sowie Simulatorvariablen lesen und schreiben.

Damit die Web-API funktioniert, muss AxisAndOhs „als Administrator“ ausgeführt werden!

21.1 URL

Der Zugriff auf die API ist über zwei Ports möglich. Die Basis-URLs für die API lauten:

http://<Lokale-IP-Adresse>:43380/webapi

http://<Lokale-IP-Adresse>:43381/webapi

<Lokale-IP-Adresse> ist entweder „localhost“ für Verbindungen auf demselben Computer oder die lokale IP-Adresse.

Beide Ports sind funktional identisch. Sie können jeden der beiden verwenden, es ist jedoch ratsam, die Last auf beide zu verteilen, wenn mehrere Programme oder Add-ons auf die API zugreifen. Der sekundäre Port ist immer der primäre Port plus 1.

Beachten Sie Firewall-Einschränkungen. Möglicherweise müssen Sie die TCP-Ports in Ihrer lokalen Firewall oder der Router-Firewall öffnen.

21.2 Ändern der Ports

Wenn in Ihrem lokalen Netzwerk andere Anwendungen den Port 43380/43381 nutzen, müssen Sie diesen ändern. Die Portänderung muss immer an beiden Enden erfolgen, sowohl in AAO als auch in der App/Webseite, die mit AAO kommunizieren möchte.

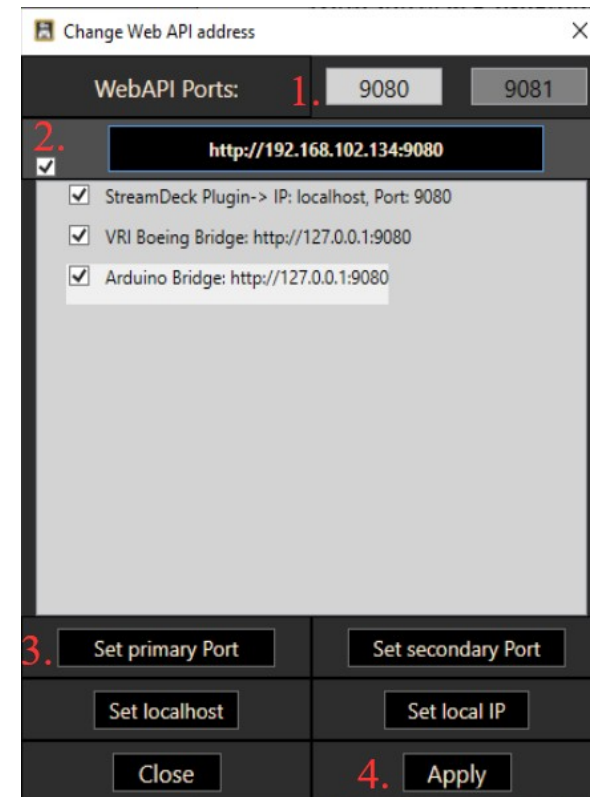
Bei AAO kann der Port über „**Tools** → **Configure WebAPI Port**“ (Extras → WebAPI-Port konfigurieren) geändert werden. AxisAndOhs muss neu gestartet werden, wenn Sie den Port ändern.

In diesem Dialogfeld können Sie den Port mit dem Mausrad einstellen.

Die Liste zeigt alle auf Ihrem Computer gefundenen AAO-Add-ons an, die diesen Port verwenden. Sie können die Adresse in diesen Add-ons mit den Schaltflächen unterhalb der Liste anpassen:

Primären/sekundären Port festlegen: Wende diesen Port auf die ausgewählten Add-ons an.
Localhost/lokale IP festlegen: Ändere die URL in den Add-ons zu localhost oder deiner lokalen IP-Adresse.

Bitte beachte, dass die Add-ons zu diesem Zeitpunkt nicht laufen dürfen. Sollten sie laufen, werden sie in der Liste deaktiviert. Du musst das andere Programm beenden, um die Änderung vorzunehmen.



21.3 Senden eines simulierten Schaltflächenereignisses

Sie können Schaltflächenereignisse des Controllers über Anfrageparameter senden:

`http://localhost:43380/webapi?dev=1&chn=5&btn=4`

`dev`, `chn` und `btn` sind beliebige numerische Werte, die Sie entsprechend Ihrem Anwendungsfall festlegen können. Diese Ereignisse werden in den Dialogen der Schaltflächen „Hinzufügen“ und „Ändern“ als WebAPI-Eingabe angezeigt.

- **dev:** ist die ID des „Geräts“, von dem das Ereignis gesendet wird. Diese sollte pro Web-App/Seite, die mit AAO kommuniziert, eindeutig sein
- **chn:** kann verwendet werden, um dasselbe Steuerelement (`btn`) unterschiedliche Aktionen ausführen zu lassen. Damit können Sie mehrere Aktionsebenen für dasselbe Steuerelement festlegen
- **btn:** ist das Steuerelement selbst.
- **bval:** optional: Wert der Schaltfläche, 127 für gedrückte Schaltfläche, 0 für losgelassene Schaltfläche. Sie müssen beide Werte senden, andernfalls bleibt die Schaltfläche hängen.

21.4 Senden eines simulierten Achsereignisses

Sie können Achsereignisse mithilfe von Anfrageparametern senden:

`http://localhost:43380/webapi?dev=1&chn=1&axis=4&aval=16234&dmin=0&dmax=65535`

dev, chn und axis sind beliebige numerische Werte, die Sie entsprechend Ihrem Anwendungsfall festlegen können. Diese Ereignisse werden in den Dialogen „Achse hinzufügen/ändern“ als WebAPI-Eingabe angezeigt

dev: ist die ID des „Geräts“, von dem das Ereignis gesendet wird. Diese sollte pro Web-App/Seite, die mit AAO kommuniziert, eindeutig sein

chn: kann verwendet werden, um dasselbe Steuerelement (btn) unterschiedliche Aktionen ausführen zu lassen. Damit können Sie mehrere Aktionsebenen für dasselbe Steuerelement festlegen

axis: identifiziert die einzelne Achse.

value: aktuelle Achsposition zwischen dmin und dmax.

dmin: Minimalwert für die Achse; bei einem Joystick wäre dies 0.

dmax: Maximalwert für die Achse; bei einem Joystick wäre dies 65535.

21.5 JSON-Schnittstelle

Die API kann auch eine JSON-Datenstruktur verarbeiten, die als Anfrageparameter „?json=“ übermittelt wird.

```
{
  "buttons": [
    {
      "dev": <numerical device id>,
      "chn": <numerical channel id>
      "btn": <numerical button id>
      "bval": <optional: 127 for button pressed, 0 for button released>
    }, ...
  ],
  "axis": [
    {
      "dev": <numerical device id>,
      "chn": <numerical channel id>,
      "axis": <numerical axis id>,
      "value": <numerical value>,
      "devMin": <minimum numerical value for this axis>,
      "devMax": <maximum numerical value for this axis>,
    }, ...
  ],
  "triggers": [
    {
      "evt": "<simulator event-id>",
      "value": <numerical value to set>
    }, ...
  ],
  "getvars": [
    {
      "var": "<simulator variable get definition>",
      "value": 0.0
    }, ...
  ],
  "getstringvars": [
    {
      "var": "<simulator variable get definition>",
      "value": "<string>"
    }, ...
  ],
  "setvars": [
    {
```

```

    "var": "<simulator variable set definition>",
    "value": <numerical value to set>
  }, ...
],
"setstringvars": [
  {
    "var": "<variable name, this can only be used with AAO internal LVars of type , String>",
    "value": <string value to set>
  }, ...
],
"pulllvars": [
  {
    "var": "<simulator variable get definition>",
    "value": 0.0
  }, ...
],
"pullstringlvars": [
  {
    "var": "<simulator variable get definition>",
    "value": ""
  }, ...
],
"scripts": [
  {
    "code": "<RPN script code>"
  }, ...
],
"inits": [
  {
    "code": "<RPN script code with AVars, (>K:scriptcall) or (>S:scriptcode) that should be initialized>"
  }, ...
],
"files": [
  {
    "fname": "<file path on the AAO webserver>",
    "content": "<file content as string>", [content and base64 are mutually exclusive, you can use only one of them]
    "base64": "<binary file content as Base64 encoded string>",
    "isvirtual": "true" , [optional, file is retained in memory on the server, not saved to disk]
  }, ...
]
}

```

„pull“-Anfragen geben den Wert der Variablen zurück und setzen sie gleichzeitig auf 0 bzw. eine leere Zeichenkette zurück. Sie gelten nur für AAO-interne LVars, wie beispielsweise (L:myvar) und (L:mystrvar, String)

Mit JavaScript-XMLHttpRequests würden Sie die Datenstruktur wie folgt senden:

```
var requestObj = {};  
requestObj.buttons = [];  
var tosend = { "dev": 1, "chn": 1, „btn“:1 };  
requestObj.buttons.push(tosend);  
tosend = { "dev": 1, "chn": 1, „btn“:14 };  
requestObj.buttons.push(tosend);  
requestObj.triggers = [];  
tosend = {"evt":(">K:LANDING_LIGHTS_TOGGLE)","value":1.0};  
requestObj.triggers.push(tosend);  
requestObj.getvars = [];  
tosend = {"var":"(A:AUTOPILOT ALTITUDE LOCK VAR, feet)","value":0.0};  
requestObj.getvars.push(tosend);  
tosend = {"var":"(A:AUTOPILOT VERTICAL HOLD VAR, feet/minute)","value":0.0};  
requestObj.getvars.push(tosend);  
requestObj.setvars = [];  
tosend = {"var":(">A:NAV1 OBS, degrees)","value":135.0};  
requestObj.setvars.push(tosend);  
(...)  
var xhttp = new XMLHttpRequest();  
xhttp.addEventListener("load", requestListener);  
xhttp.open("GET", "http://localhost:43380/webapi?json=" + JSON.stringify(requestObj));  
xhttp.send();
```

Bei größeren Datenmengen sollten Sie eine POST-Anfrage in Betracht ziehen:

```
var xhttp = new XMLHttpRequest();
xhttp.addEventListener("load", requestListener);
xhttp.open("POST", "http://localhost:43380/webapi/", true);
xhttp.send(JSON.stringify(requestObj));
```

Die Antwort hat denselben Aufbau, wobei die „getvars“-Felder die angeforderten Werte enthalten

```
function requestListener() {
    var commObj = JSON.parse(this.responseText);
    var apAlt = commObj.getvars[0].value;
    var apVs = commObj.getvars[1].value;
}
```

Sie können auch Werte abfragen, die von einem RPN-Skript berechnet wurden, und zwar mithilfe von

„(L:scriptgroup-scriptname)“ für numerische Werte, die von einem Skript wie diesem berechnet wurden:

„(A:INDICATED·ALTITUDE,·Feet)·10000·+“ (getvars)

„(L:scriptgroup-scriptname, String)“ für Zeichenfolgenwerte, die von einem Skript wie diesem erzeugt werden:

„%(A:INDICATED·ALTITUDE,·Feet)·10000·+%!5d!“ (getStringvars)

und für ein externes Skript, das nicht in AAO gespeichert sein soll, mit dem Header „S:“

„(S:(A:INDICATED·ALTITUDE,·Feet)·10000·+)“ (getvars)

„(S:%(A:INDICATED·ALTITUDE,·Feet)·10000·+%!5d!)“ (getStringvars)

Das Beispiel in diesem Kapitel führt Folgendes aus:

- Es sendet zwei Tastenklicks, 1 und 14.
- Es schaltet die Landefeuer ein und aus und versetzt den Simulator anschließend in den Pausenmodus.
- Es ruft die vom Autopiloten gewählte Flughöhe, die Sinkgeschwindigkeit und den Kurs ab.
- Schließlich setzt es den NAV-1-OBS-Marker auf 135 Grad.

```
{
  "buttons": [
    {
      "dev": 1,
      "chn": 1,
      "btn": 1
    },
    {
      "dev": 1,
      "chn": 1,
      "btn": 14
    }
  ],
  "triggers": [
    {
      "evt": "(>K:PAUSE_ON)",
      "value": 1.0
    },
    {
      "evt": "(>K:LANDING_LIGHTS_TOGGLE)",
      "value": 1.0
    }
  ],
  "getvars": [
    {
      "var": "(A:AUTOPILOT_ALTITUDE_LOCK_VAR, feet)",
```

```
        "value": 0.0
    },
    {
        "var": "(A:AUTOPILOT VERTICAL HOLD VAR, feet/minute)",
        "value": 0.0
    },
    {
        "var": "(A:AUTOPILOT HEADING LOCK DIR, degrees)",
        "value": 0.0
    }
],
"setvars": [
    {
        "var": "(>A:NAV1 OBS, degrees)",
        "value": 135.0
    }
],
"setstringvars": [
    {
        "var": "(>L:AAO_Internal_State, String)",
        "value": "READY"
    }
]
}
```

22 Verwendung der API als Webserver

Die WebAPI kann auch als einfacher Webserver zum Hosten von HTML-Seiten verwendet werden, beispielsweise um ein Schaltflächenfeld oder Fluginstrumente bereitzustellen.

1. Erstellen Sie einen Unterordner Ihrer Wahl in \Dokumente\LorbyAxisAndOhs Files\WebPages
Beispiel: \Dokumente\LorbyAxisAndOhs Files\WebPages\mybuttonwebpage
2. Kopieren Sie dann Ihre HTML-, CSS-, JS-Dateien und Bilder in diesen Unterordner
Beispiel: \Dokumente\LorbyAxisAndOhs Files\WebPages\mybuttonwebpage\index.html
 \Dokumente\LorbyAxisAndOhs Files\WebPages\mybuttonwebpage\buttonlayout.css
 \Dokumente\LorbyAxisAndOhs Files\WebPages\mybuttonwebpage\background.png
3. Nachdem Sie AAO gestartet haben, können Sie wie folgt von jedem Browser aus auf Ihre Seite zugreifen:
<http://localhost:43380/webapi/mybuttonwebpage/index.html>
(für den Fernzugriff muss „localhost“ durch die tatsächliche IP-Adresse des AAO-Computers ersetzt werden)

Es gibt einen speziellen Code, mit dem Sie AAO dazu bringen können, die tatsächliche API-Adresse in Ihren HTML- oder JavaScript-Code einzufügen:

<AAO_URL>. Dies wird typischerweise verwendet, wenn Ihre Webseite aktive Elemente enthält, die einen Rückruf an AAO auslösen.

```
...
<body>
<script>
var AAO_URL = „<AAO_URL>“;
var deviceId = 235467;
...
xhttp.open („GET“, AAO_URL + „?json=“ + JSON.stringify(mainLoopRequestObj));
```

Auf diese Weise müssen Sie sich in Ihrem Code nicht um die tatsächliche Adresse kümmern.

Um den Zugriff auf die API im Code zu vereinfachen, können Sie eine Verknüpfung zu einer speziellen AAO-JavaScript-Bibliothek herstellen:

```
<!DOCTYPE html>
<html>
<head>
<script src="AaoWebApi.js"></script>
```

Dann erstellst du in deinem Code ein neues API-Objekt:

```
<body style="background: black">
...
<script>
  var webApi = new AaoWebApi();
...
  webApi.StartAPI(75);
```

Der letzte Befehl startet den Datenerfassungsdienst der API; in diesem Fall werden alle 75 ms neue Daten abgerufen.

Das webApi-Objekt verfügt über die folgenden Methoden, die Sie verwenden können:

- StartAPI(frequency): Startet die Datenerfassung alle frequency Millisekunden
- StopAPI(): Stoppt die Datenerfassung
- GetSimVar(varname); Ruft den Wert einer bestimmten Simulatorvariablen ab
`var val = webApi.GetSimVar(„(L:FMC_EXEC_ACTIVE, Number)“);`
- PullSimVar(varname); den Wert abrufen und auf 0/leer zurücksetzen
`var val = webApi.PullSimVar(„(L:b747fmc/fmc.html_complete)“);`
- ClearSimVar(varname): eine Variable aus der Datenerfassung entfernen
- Reset(): alle Variablen löschen

- **SetVariable(Wert, Variablenname):** Setzt eine Variable auf den angegebenen Wert
`webApi.SetVariable(1, `(>L:AAO_FMC_HTML_INIT, Number) `);`
- **SendEvent(Wert, Ereignis):** Löst ein Ereignis in AAO aus
`webApi.SendEvent(0, `(>H:B747_8_FMC_1_BTN_PLUSMINUS, Zahl) `);`
- **SendButton(Geräte-ID, Kanal-ID, Tasten-ID):** Senden eines Tastenereignisses
`webApi.SendButton(365, 10, 2);`
- **SendScript(Code):** Senden eines RPN-Codes zur Verarbeitung an AAO
`webApi.SendScript(`1 (L:Statevar) - (>L:Statevar) `);`

Die WebAPI kann auch zum Erstellen von Textdateien auf dem Webserver verwendet werden.

Die Struktur für die Übertragung einer Datei in der JSON-Anfrage lautet

```
{
"fname": "/testfolder/testfile.txt",
"content": "this it the text content of the file",
"base64": "this it the binary content of the file"
"isvirtual": "true"
}
```

Sie können entweder <content> oder <base64> verwenden, jedoch nicht beides. Jede Struktur, die ein <content> enthält, wird als Textdatei geschrieben, jede Datei, die ein <base64> enthält, wird als Binärdatei geschrieben. „isvirtual“ ist optional; wenn es vorhanden und auf „true“ gesetzt ist, wird die Datei nicht auf der Festplatte des Servers gespeichert, sondern nur im Arbeitsspeicher gehalten.

```
var requestObj = {};
requestObj.files = [];
var tosend = { "fname": "/testfolder/testfile.txt", "content": "this it the content of the file" };
requestObj.files.push(tosend);
var tosend = { "fname": "/testfolder/testing.jpg", "base64": "iVBORw0KGgoAAAANSUHE...." };
requestObj.files.push(tosend);
(...)
var xhttp = new XMLHttpRequest();
xhttp.open("POST", "http://localhost:43380/webapi?json=" + JSON.stringify(requestObj));
xhttp.send();
```

Dies wird in der Regel verwendet, um externe HTML-Dateien oder Teile davon an eine Website auf dem AAO-Server zu senden, wo sie zur Laufzeit eingebunden werden sollen. Dies funktioniert auch, wenn Sie Daten aus dem Simulator heraus senden, wobei dieselbe XMLHttpRequest-Logik wie oben gezeigt zum Einsatz kommt.

Webseiten aufrufen

Sobald AxisAndOhs mit dem Simulator verbunden ist oder im Offline-Modus läuft, können Sie über den Dialog unter „Gauges->Display WebPages“ die konfigurierten Webseiten aufrufen und deren URLs abrufen

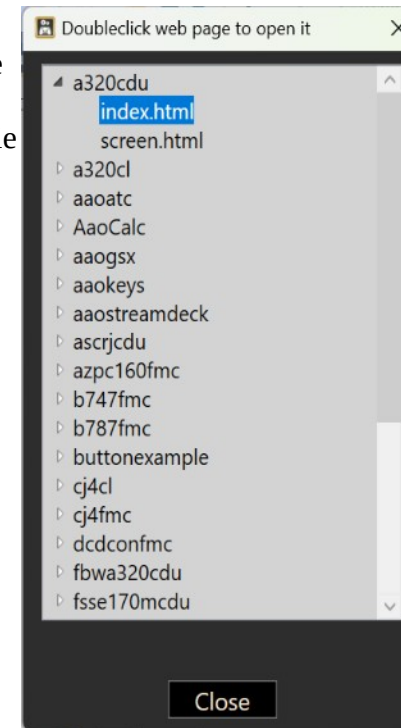
- *Doppelklicken* Sie auf eine HTML-Seite, um sie in Ihrem Standardbrowser zu öffnen
- *Klicken Sie mit der linken Maustaste und anschließend mit der rechten Maustaste* auf eine HTML-Seite, um die URL dieser Seite in die Windows-Zwischenablage zu kopieren. Sie können sie dann mit Strg + V in die Adressleiste Ihres Browsers einfügen.

Fertige Beispiele für AAO-Webseiten können unter

<https://www.axisandohs.com/downloads.html>

heruntergeladen werden.

Dazu gehören mehrere FMC/CDU-Instrumente für verschiedene Flugzeuge, eine einfache Schaltflächenseite, eine einfache Moving Map und vieles mehr.



23 Ereignis- und Variablenlisten importieren

Über „Scripting → Import Events and Variables“ (Skripting → Ereignisse und Variablen importieren) können Sie CSV-Textdateien mit Listen von Ereignissen und Variablen in eine benutzerdefinierte Gruppe importieren. Über „Skripting → Benutzerdefinierte Ereignisse und Variablen bearbeiten“ können Sie diese wieder löschen.

Das Dateiformat ist ein einfacher CSV-Dateiformat („Comma Separated Values“), das mit jedem Texteditor erstellt werden kann.

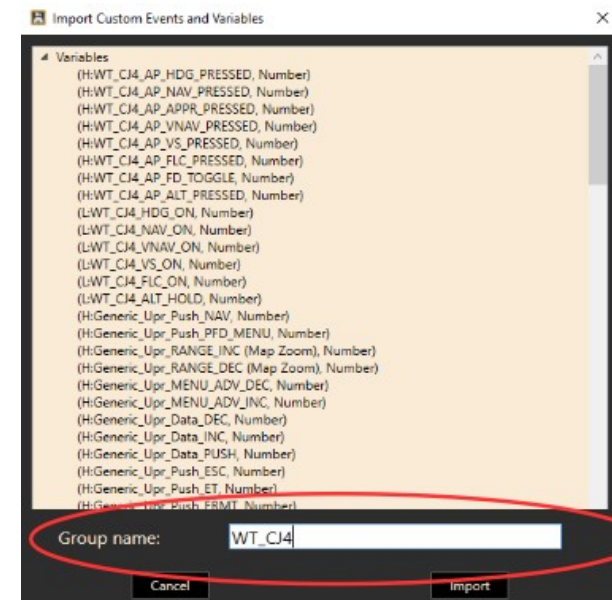
- Für ein Ereignis lautet die Syntax
`K,MeineEigenesEreignisID[,Beschreibung][,Gruppe]`
(nur K- und B-Ereignisse sind zulässig, und die Ereignis-ID darf keine Leerzeichen enthalten)
- für eine Variable lautet es
`A,MeinEigenesVariablenname,Variableneinheit[,Beschreibung][,Gruppe]`
nur A-, L-, H-, E-, P- und C-Variablen sind zulässig;

„Beschreibung“ und „Gruppe“ sind optional.

Beispiel für eine Importdatei (myfile.csv):

```
K,My_Own_Kevent_1
K,My_Own_Kevent_2
K,#76554,This is my Event 3,My Events
A,Undocumented Headingvar,Degrees
L,MyOwnLVar,Number
H,MyOwnHVar,Number
H,AnotherHVar,Number
```

Geben Sie vor dem Import einen Gruppennamen ein oder geben Sie ihn in der Datei an



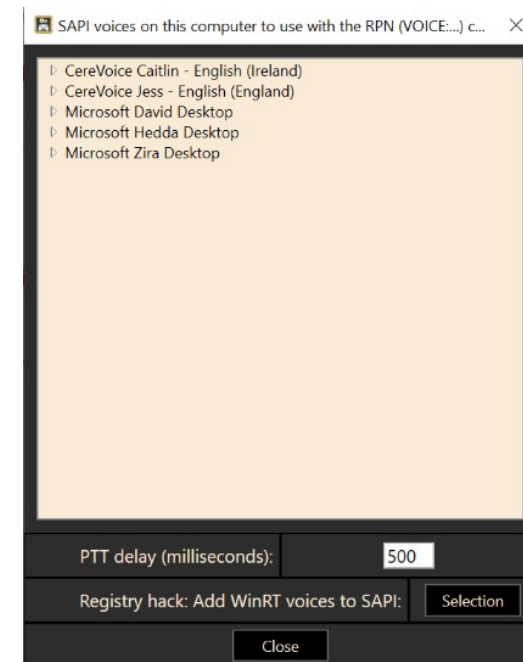
24 Text-to-Speech: WinRT vs. SAPI

Wenn Sie die Befehle (SPEAK: und (VOICE: verwenden, werden Sie feststellen, dass unter Windows 10 nicht alle Stimmen, die Ihnen für Text-to-Speech zur Verfügung stehen, auch im AAO-Dialogfeld „Extras -> Liste aller SAPI-Stimmen anzeigen“ sichtbar sind.

Der Grund dafür ist, dass Windows tatsächlich über zwei Sprachsysteme verfügt: „WinRT“ und „SAPI“. SAPI ist die ältere, bekanntere Methode. Stimmen, die Sie online kaufen können, liegen meist im SAPI-Format vor.

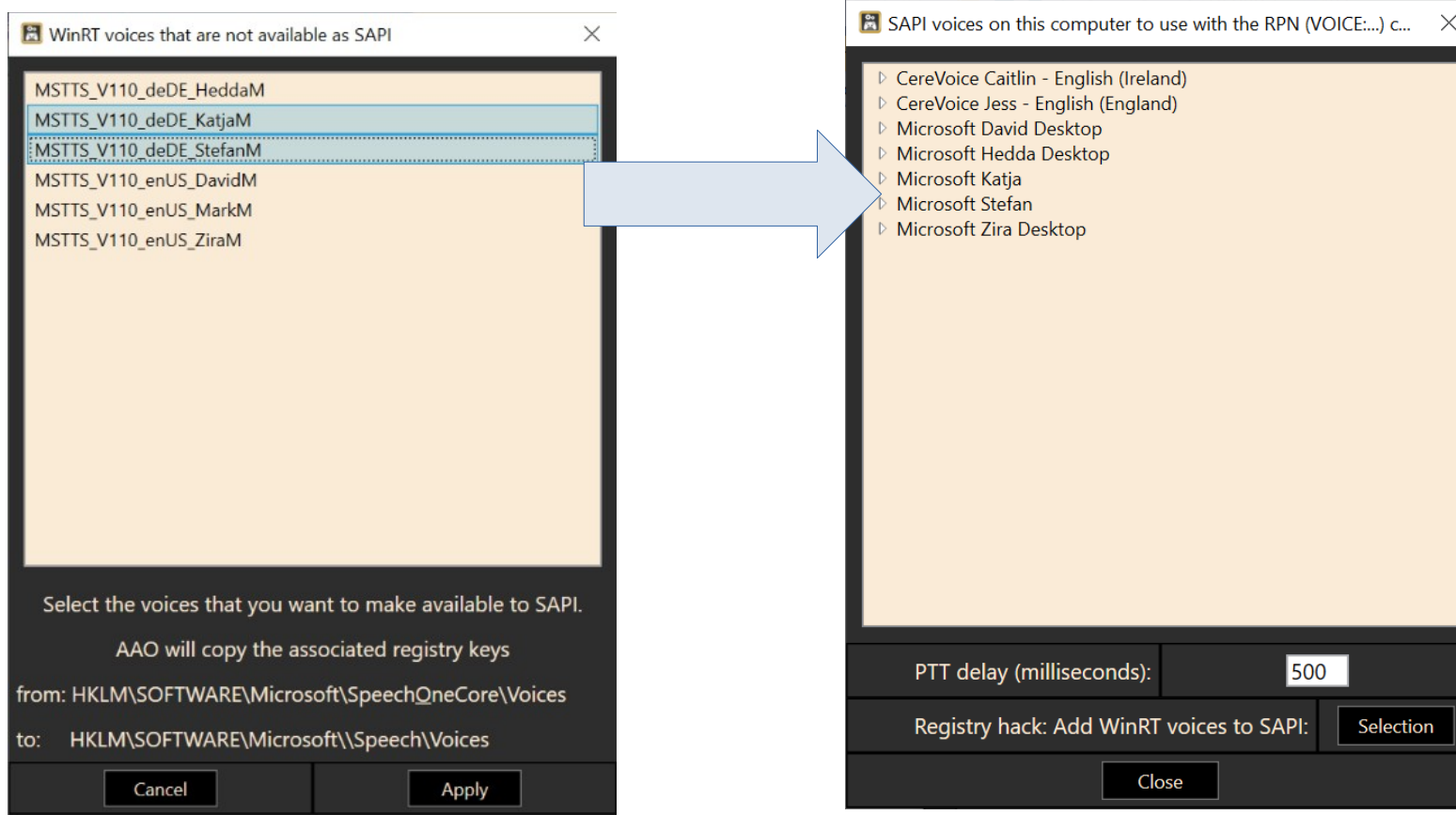
Es ist möglich, die WinRT-Stimmen auch als SAPI zu verwenden, doch dafür müssen eine Reihe von Registrierungsschlüsseln an einen anderen Speicherort dupliziert werden.

AAO kann dies für Sie übernehmen, und zwar über die Schaltfläche „Auswahl“ am unteren Rand des Dialogfelds. Diese Schaltfläche ist nur aktiv, wenn AAO „Als Administrator“ gestartet wurde und wenn tatsächlich zusätzliche WinRT-Stimmen vorhanden sind, die Sie verwenden können.



Wenn Sie auf die Schaltfläche klicken, wird eine Liste der verfügbaren WinRT-Stimmen angezeigt. Wählen Sie diejenigen aus, die Sie als SAPI verwenden möchten, und klicken Sie dann auf „Ausführen“.

Nach Abschluss des Vorgangs müssen Sie Ihren Computer neu starten (!), da die Änderungen an der Registrierung sonst nicht wirksam werden.



25 Erweiterte Text-to-Speech-Funktionen: Azure, Polly, ChatGPT

Wenn Sie über die Funktionen von Windows SAPI hinausgehen möchten, können Sie die Dienste von Microsoft Cognitive Speech („Azure“) und Amazon AWS Polly nutzen. Kopieren Sie einfach den Namen einer Stimme aus einem der Dienste und fügen Sie ihn in den Befehl (VOICE: in Ihrem Skript ein.

Darüber hinaus verfügt AxisAndOhs über eine Schnittstelle zur ChatGPT-API, über die Sie Text generieren können.

Beachten Sie, dass Sie für alle diese Dienste einen API-Schlüssel vom Anbieter benötigen und dass einige davon kostenpflichtig sind.

Ihre API-Schlüssel müssen über die Dialoge im Menü „Extras“ eingegeben werden.

Microsoft Cognitive Speech („Azure“)

Azure kann für die Sprachausgabe genutzt werden. Es bietet eine große Auswahl an verschiedenen Stimmen für viele Sprachversionen.

[Erstellen Sie noch heute Ihr kostenloses Azure-Konto | Microsoft Azure](#)

Sobald Sie Ihr Konto erstellt haben, fügen Sie die Ressource „azure-speech-api“ hinzu und erstellen Sie anschließend den API-Schlüssel dafür. Der Azure-Dienst ist derzeit kostenlos, solange Ihre Nutzung unter dem vordefinierten Limit von 500.000 Zeichen pro Monat bleibt.

Nachdem Sie Ihre API-Schlüssel in AxisAndOhs eingegeben haben, können Sie über „Extras->Show list of Microsoft Azure Voices“ (Extras->Liste der Microsoft Azure-Stimmen anzeigen) auf eine Liste der verfügbaren Stimmen zugreifen. Um den Namen einer Stimme zu kopieren, klicken Sie mit der linken Maustaste darauf und dann mit der rechten Maustaste. Dadurch wird der Name in die Windows-Zwischenablage kopiert.

Fügen Sie den Namen der Stimme wie folgt in Ihren Skriptcode ein:

```
(VOICE:Azure|en-US-AndrewNeural)
```

und der nächste SPEAK/SPEAKBLK-Befehl wird von Azure gesprochen.

Amazon AWS Polly

AWS Polly kann für die Sprachausgabe verwendet werden. Es bietet zwar nicht so viele Stimmen wie Azure, die Verarbeitung ist jedoch schneller.

[AWS-Konsole – Anmeldung \(amazon.com\)](#)

Nachdem Sie Ihr Konto erstellt haben, legen Sie über IAM eine neue Gruppe an und fügen Sie „AmazonPollyFullAccess“ als Berechtigungsrichtlinie hinzu. Erstellen Sie anschließend einen neuen Benutzer in dieser Gruppe. Nun können Sie einen API-Schlüssel für diesen Benutzer erstellen. Der AWS-Dienst ist derzeit für 12 Monate kostenlos. Bitte konsultieren Sie die AWS-Preistabellen für die weitere Nutzung.

Sie können eine Liste der verfügbaren Stimmen unter „Extras->Liste der Amazon Polly-Stimmen anzeigen“ aufrufen. Um den Namen einer Stimme zu kopieren, klicken Sie mit der linken Maustaste darauf und dann mit der rechten Maustaste. Dadurch wird der Name in die Windows-Zwischenablage kopiert.

Fügen Sie den Namen der Stimme wie folgt in Ihren Skriptcode ein:

(VOICE:Polly|en-US Danielle|neural)

und der nächste SPEAK/SPEAKBLK-Befehl wird von AWS Polly gesprochen.

Chat GPT

Sie können auch Ihr ChatGPT-API-Konto (siehe separates Kapitel weiter unten) verwenden, um den Text von der KI vorlesen zu lassen. ChatGPT bietet nur eine begrenzte Auswahl an Stimmen und ist im Vergleich zu den anderen Diensten langsam. Der ChatGPT-API-Dienst ist kostenpflichtig.

Eine Liste der verfügbaren Stimmen finden Sie unter „Extras->Liste der ChatGPT-Stimmen anzeigen“. Um den Namen einer Stimme zu kopieren, klicken Sie mit der linken Maustaste darauf und dann mit der rechten Maustaste. Dadurch wird der Name in die Windows-Zwischenablage kopiert.

Fügen Sie den Namen der Stimme wie folgt in Ihren Skriptcode ein:

```
(VOICE:GPT|alloy)
```

und der nächste SPEAK/SPEAKBLK-Befehl wird von ChatGPT vorgelesen.

26 RPN-Skriptdateien

Wenn Ihre RPN-Skripte zu komplex werden oder Sie mehrere Skripte mit einem einzigen Befehl ausführen möchten, sollten Sie die Verwendung von Skriptdateien in Betracht ziehen.

Eine Skriptdatei ist eine einfache Textdatei, in der jede Zeile ein RPN-Skript darstellt. Die Skripte können so einfach oder komplex sein, wie Sie möchten, aber Sie müssen darauf achten, dass jedes Skript immer nur eine einzige Zeile einnimmt:

```
(A:PLANE HEADING DEGREES GYRO, Degrees) (>K:HEADING_BUG_SET)
(FOCUS:outlook) · (VKD:28-156-13) · (SPLIT:100) · (VKU:28-156-13) · (SPLIT:5000) · (FOCUS:prepar3d)
53623 · (>L:MJC_VAR_READ_CODE, ·Number) · (L:MJC_VAR_READ_VALUE, ·Number) · (>L:MJC_APU_PWR)
0 (>K:VIEW_REAR)
0 (>K:VIEW_REAR)
0 (>K:VIEW_REAR)
0 (>K:VIEW_REAR)
0 (>K:VIEW_REAR)
...
```

Alle Skriptdateien sind im Ordner „\Dokumente\LorbyAxisAndOhs Files\Scripts“ zu speichern (Sie können Unterordner anlegen)

Skriptdateien ausführen

Es gibt zwei Möglichkeiten, eine Skriptdatei auszuführen:

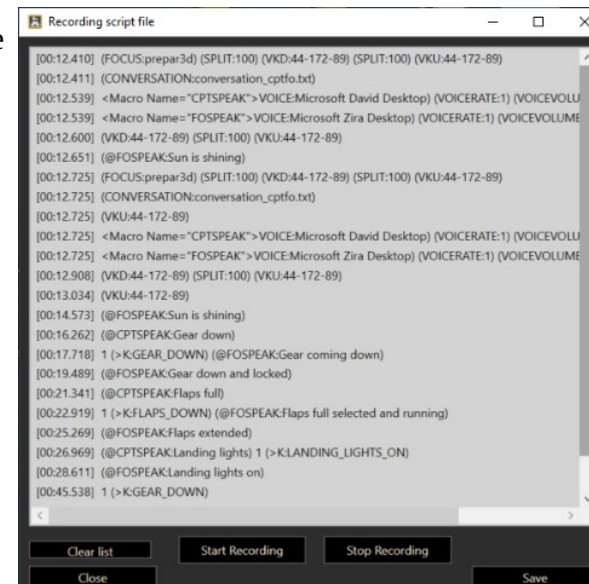
- entweder manuell über das Dialogfeld unter „Scripting → Skriptdateien ausführen“
- oder durch Aufruf innerhalb eines Skripts mit dem Befehl „(SCRIPTFILE:Dateiname)“ (Spezifikationen siehe Liste der AAO-Befehle oben). Dem Dateinamen muss der Name des Unterordners vorangestellt werden, falls vorhanden:
(SCRIPTFILE:myfolder\myfile.txt) (=der gesamte Pfad relativ zu \Scripts\)

Aufzeichnen von Skriptdateien

AAO kann alle Eingabeaktionen aufzeichnen, die Sie mit der App ausführen, alle aufgerufenen Skripte und, wenn Sie gleichzeitig den Ereignisbeobachter „Scripting->Watch simulator events“ öffnen, auch die Simulatorereignisse, die innerhalb des Simulators selbst ausgelöst werden.

Es gibt zwei Möglichkeiten, Aktionen aufzuzeichnen:

1. Verwendung der Befehle AAO_RECORDING_ON / AAO_RECORDING_OFF
Dadurch wird automatisch eine Skriptdatei im Skriptordner erstellt
2. Verwendung des Dialogs unter „Scripting->Record script files“
 - In diesem Dialog können Sie die Aufzeichnung starten und stoppen und die eingehenden Ereignisse beobachten
 - Sie können die Aufzeichnung in einer beliebigen Datei speichern, sollten diese jedoch im Skriptordner ablegen
 - Beim Speichern fragt die App, ob Sie die Zeitcodes einbeziehen möchten, damit die Ereignisse zum gleichen Zeitpunkt wie in Ihrer Aufzeichnung ausgelöst werden oder nicht.
 - Wenn Sie Ereignisse aufzeichnen möchten, die vom Simulator stammen, müssen Sie gleichzeitig den Dialog „Ereignisbeobachter“ öffnen (es gelten alle Regeln zum Ignorieren von Ereignissen usw.)



27 Interaktive Text-to-Speech-Checkliste

AAO verfügt über eine fest integrierte Funktion zum Ausführen interaktiver Checklisten. Checklisten sind RPN-Skriptdateien in einem speziellen Format.

Jede Zeile in der Datei, die mit [] (öffnende + schließende Klammer) beginnt, wird als Teil einer Checkliste betrachtet, und AAO wechselt an dieser Stelle in den Checklistenmodus. Checklisten werden unter „Dokumente\LorbyAxisAndOhs Files\Scripts“ gespeichert.

Beispiel:

Erstellen Sie eine Datei: \Dokumente\LorbyAxisAndOhs Files\Scripts**a320_before_start_cl.txt**

```
<Macro Name="FOSPEAK">VOICE:Microsoft Zira Desktop) (VOICERATE:0) (VOICEVOLUME:75) (SPEAK</Macro>
(@FOSPEAK:Before start checklist) (WAIT:4000)
[](@FOSPEAK:fueling)
(@FOSPEAK:checked)
[](@FOSPEAK:chocks)
(@FOSPEAK:removed)
[](@FOSPEAK:traffic cones)
(@FOSPEAK:clear)
[](@FOSPEAK:A P U)
(@FOSPEAK:checked on)
[](@FOSPEAK:A P U bleed)
(@FOSPEAK:checked on)
[](@FOSPEAK:external power)
(@FOSPEAK:disconnected)
[](@FOSPEAK:doors)
(@FOSPEAK:closed)
[](@FOSPEAK:beacon)
(@FOSPEAK:checked on)
(WAIT:1500) (@FOSPEAK:before start checklist complete, pushback is next)
```

Funktionsweise der Checkliste:

Die Checkliste wird durch den Aufruf von **(CHECKLIST:a320_before_start_cl.txt)** in einem RPN-Skript gestartet.

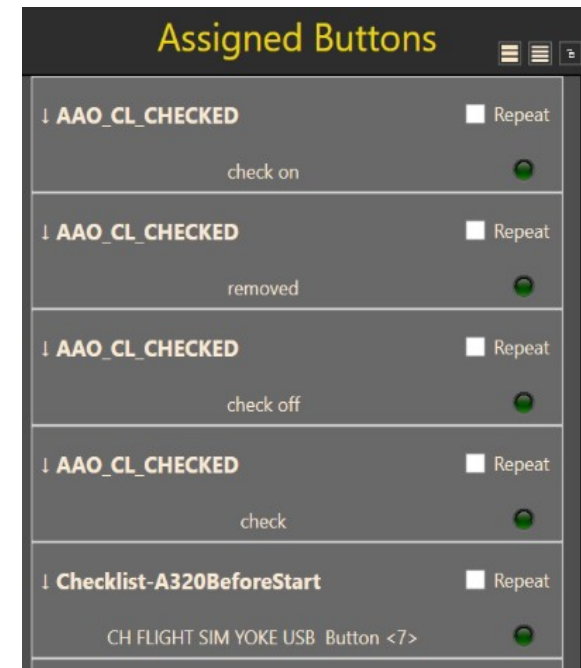
Bei jeder Zeile, die mit [] beginnt, führt AAO den Rest der Zeile aus und wartet anschließend auf Ihre Bestätigung.

Standardmäßig erfolgt die Bestätigung durch Auslösen des Ereignisses „Audio Checklist → AAO_CL_CHECKED“ mit einer Standard-AAO-Tastenbelegung. Dies kann mit jedem Eingabegerät erfolgen, einschließlich Spracherkennung.

Wenn Sie nicht innerhalb von 10 Sekunden bestätigen, wird der Befehl automatisch wiederholt.
Wenn Sie mehr als 5 Mal nicht bestätigen, wird die Checkliste gestoppt.

Die Klammern [] können zusätzliche Verarbeitungsinformationen enthalten:

- [0]: AAO wartet nicht auf eine Bestätigung. Siehe untenstehendes Beispiel für ein Gespräch zwischen Kapitän und Erster Offizier.
- [Anzahl der ms]: ändert den Wert für das Zeitlimit der Bestätigung. „[15000]“ bedeutet, dass Sie nach 15 Sekunden um Bestätigung gebeten werden. Sie können keinen Wert unter 5000 ms festlegen (5 Sekunden)
- [*Phrase*]: benutzerdefinierte Sprachbestätigung – siehe unten
- [RPN-Code] bewirkt, dass die Checkliste an dieser Stelle wartet, bis die Bedingung im RPN-Code erfüllt ist (siehe Beispiel auf der nächsten Seite)
- [RPN-Code|SKIP] überspringt diese Zeile, wenn die Bedingung nicht erfüllt ist



Steuerung des Checklistenablaufs

Die Ereignisse zur Ablaufsteuerung befinden sich im Dialogfeld zur Ereignisauswahl in der Gruppe „Audio-Checkliste“.

AAO_CL_CHECKED: Checkliste fortsetzen

AAO_CL_STOP: Checkliste anhalten

AAO_CL_PAUSE_TOGGLE: Checkliste unterbrechen, bis das Ereignis erneut empfangen wird.

Es stehen drei LVars zur Verfügung, die für die Ablaufsteuerung der Checkliste verwendet werden können. Sie können jederzeit gelesen und beschrieben werden. Wenn Sie also eine andere Zeitüberschreitung wünschen, setzen Sie die LVar einfach gleich zu Beginn Ihrer Checklisten-Datei auf einen neuen Wert.

(L:AaoClRepeatMs): enthält die Zeit in Millisekunden, bis eine wartende Checkliste beginnt, die Wiederholungsaktion auszuführen

(L:AaoClTimeout): enthält die Anzahl der Wiederholungsaktionen, die fehlschlagen müssen, bevor die Checkliste gestoppt wird

Sowohl für CHECKLIST als auch für CONVERSATION können Sie die aktuell aktive Zeile mit (L:filename.txt) abfragen.

Der Wert „0“ bedeutet, dass die Skriptdatei nicht aktiv ist.

Beispiel: (L:conversation_cptfo.txt)

Benutzerdefinierte Wiederholung

Wenn Sie möchten, dass nach Ablauf der Wartezeit der Checkliste eine andere Aktion ausgeführt wird, fügen Sie vor dem Checklistenpunkt eine Zeile ein, die mit einem „*“ beginnt und endet und die gewünschte Aktion enthält:

```
*(VOICE:Microsoft Zira Desktop) (VOICERATE:0) (VOICEVOLUME:75) (SPEAK:Bitte überprüfen Sie es erneut)*  
[] (@FOSPEAK:A P U)
```

Benutzerdefinierte Bestätigung (nur Spracherkennung)

Parallel zum standardmäßigen Bestätigungsereignis (AAO_CL_CHECKED) können Sie im Bestätigungsfeld des Checklistenelements Listen mit Phrasen verwenden, die Sie selbst aussprechen möchten, um die Checkliste fortzusetzen. Dies funktioniert nur bei ordnungsgemäß eingerichteter Windows-Spracherkennung; siehe auch das Kapitel über Spracherkennung.

```
[yes|no|maybe|whatever] rpncode
```

Nachdem eine der Phrasen empfangen wurde, wird die Checkliste weitergeschaltet und (L:AaoClConfirm) enthält den Index der erkannten Phrase, beginnend mit „1“. Das Standard-Bestätigungsereignis hat den Index „0“.

Beispiel:

```
<Macro Name="COVOICE">VOICE:Microsoft Mark) (VOICERATE:0) (VOICEVOLUME:100) (SPEAK</Macro>
*(VOICE:Microsoft David Desktop) (VOICERATE:0) (VOICEVOLUME:100) (SPEAK:Please recheck)*
(@COVOICE: approach checklist) (WAIT:2000)
:redo
[*completed|no|maybe*](@COVOICE: approach briefing)
(L:AaoClConfirm) 0 == if{ (@COVOICE:checked) (GOTO:cont) }
(L:AaoClConfirm) 1 == if{ (@COVOICE:confirmed) (GOTO:cont) }
(L:AaoClConfirm) 2 == if{ (@COVOICE:why not?) }
(L:AaoClConfirm) 3 == if{ (@COVOICE:what is maybe supposed to mean?) }
(GOTO:redo)
:cont
[](@COVOICE: ecam status)
(@COVOICE:checked)
...
```

Konversationsmodus

Sie können „Konversationen“ auch ausführen, indem Sie eine Datei im Checklistenformat mit (CONVERSATION:Dateiname.txt) aufrufen. Für eine Konversation ist keine Benutzerinteraktion erforderlich oder vorgesehen; die Liste wird einfach Zeile für Zeile abgearbeitet.

Sie können mehrere Konversationsstränge gleichzeitig ausführen, jedoch nur eine Checkliste.

In diesem Beispiel „conversation_cptfo.txt“ erteilt der Kapitän Befehle zur Bedienung der Flugzeugsysteme, die der Erste Offizier dann ausführt und bestätigt. Es ist keine Interaktion erforderlich.

```
<Macro Name="CPTSPEAK">VOICE:Microsoft David) (VOICERATE:1) (VOICEVOLUME:95) (SPEAKBLK</Macro>
<Macro Name="FOSPEAK">VOICE:Microsoft Zira) (VOICERATE:1) (VOICEVOLUME:75) (SPEAKBLK</Macro>
;conversation example
[](@CPTSPEAK:Gear down)
[] 1 (>K:GEAR_DOWN) (@FOSPEAK:Gear coming down)
[(A:GEAR TOTAL PCT EXTENDED, Percent) 0.99 >] (@FOSPEAK:Gear down and locked)
[](@CPTSPEAK:Flaps full)
[] 1 (>K:FLAPS_DOWN) (@FOSPEAK:Flaps full selected and running)
[(A:TRAILING EDGE FLAPS LEFT PERCENT, Percent over 100) 0.99 >] (@FOSPEAK:Flaps extended)
[](@CPTSPEAK:Landing lights) 1 (>K:LANDING_LIGHTS_ON)
[(A:LIGHT LANDING ON, Bool)] (@FOSPEAK:Landing lights on)
```

Um eine **Konversation zu beenden**, kannst du den AAO-Befehl „(AAO_CV_STOP:Dateiname.txt)“ verwenden.

Um eine **Konversation zu unterbrechen**, kannst du den AAO-Befehl „(AAO_CV_PAUSE_TOGGLE:Dateiname.txt)“ verwenden.

28 Der Dialog „Checklist Creator“

Über „Extras -> Checklist Creator“ können Sie mithilfe eines Editor-Dialogs Audio-Checklisten erstellen, einschließlich einer webbasierten Oberfläche – ganz ohne Programmierkenntnisse.

The image illustrates the workflow of the Checklist Creator tool through three sequential screenshots:

- Checklist Creator Editor:** A desktop application window titled "Checklist Creator" showing a list of checklist items categorized into sections like "PREFLIGHT", "SAFETY CHECK", and "POWER UP". Each item has a checkbox and a response code (e.g., "ON", "RET", "OFF", "IDLE"). A "Speed Brake Lever" item is highlighted. The interface includes buttons for "New Section", "New Item", "Import", "Save", "Load", "Delete", and "Exit".
- Checklist Sections Web Interface:** A browser window titled "a330cl Checklist" displaying a list of checklist sections: "PREFLIGHT", "SAFETY CHECK", "POWER UP", "APU START", and "OVERHEAD PANEL". Each section has a checkbox.
- SAFETY CHECK Web Interface:** A browser window titled "SAFETY CHECK" showing a detailed checklist of items for the "SAFETY CHECK" section. The items are: "Parking Brake -- ON", "Speed Brake Lever -- RET", "Flaps Lever -- UP", "Engine Start Selector -- NORM", "Engine Master 1 + 2 -- OFF", "Thrust Levers -- IDLE", "Weather Radar SYS + PWS -- OFF", "Landing Gear Lever -- DOWN", and "Wipers -- OFF". A "Start SAFETY CHECK" button is visible at the top. Below the checklist, there are buttons for "Next: POWER UP", "CHECKED", and "STOP".

Checklisten können entweder nur mit der Maus oder mithilfe des AAO-Checklistensystems „Challenge and Response“ bedient werden. Wenn Sie einen Checklistenabschnitt mit der grünen Schaltfläche starten, liest AAO die Punkte in der von Ihnen ausgewählten Stimme vor und wartet dann darauf, dass Sie dies mit dem Ereignis (K:AAO_CL_CHECKED) bestätigen, das in AAO einer Schaltfläche oder einem Sprachbefehl zugewiesen werden kann.

Checklisten-Kurzname

Dies ist der wichtigste Teil der Checkliste. Der Kurzname bestimmt, wie die Checklisten-Assets auf Ihrer Festplatte gespeichert werden und im Web zugänglich sind.

- \Documents\LorbyAxisAndOhs Files\WebPages*.*
- \Documents\LorbyAxisAndOhs Files\Scripts*.*
- http://localhost:43380/webapi/<Kurzname>/index.html

Baumansichtsfenster

Eine Checkliste ist in „Abschnitte“ unterteilt, die jeweils „Elemente“ enthalten. Ein „Abschnitt“ entspricht einer einzelnen Checkliste, die gemäß dem SOP-Segment vollständig ausgefüllt werden muss. Jeder Abschnitt enthält mehrere Elemente, die entweder abhakbar sind oder lediglich Informationen darstellen. Das Kontrollkästchen neben jedem Element bestimmt, um welche Art es sich handelt:

- Kontrollkästchen ist „aktiviert“ => dies ist ein reguläres Element der Challenge/Response-Checkliste
- Kontrollkästchen ist „deaktiviert“ => diese Zeile dient nur zur Information und wird lediglich auf der Webseite angezeigt.

Element-Editor

Wenn Sie auf ein Element klicken, wird dessen Inhalt in die Editorfelder unten übernommen. Wenn Sie den Text dort ändern, wird der Text im Element sofort angepasst.

Das Textfeld „TTS-Ersetzung“ kann verwendet werden, um das Text-to-Speech-System dazu zu bringen, den Checklistenpunkt korrekt auszusprechen (zum Beispiel „addeers“ statt „ADIRs“)

Elemente verschieben

Checklistenelemente können mit den Schaltflächen „Links“ und „Rechts“ „hochgestuft“ und „herabgestuft“ sowie nach oben und unten verschoben werden.

Speichern und Laden

Mit „Save“ (Speichern) und „Load“ (Laden) wird die aktuelle Version in einer Datei auf Ihrer Festplatte gespeichert.

Checkliste erstellen

Die Schaltfläche „Checkliste erstellen“ ist der letzte Schritt – sie erstellt alle erforderlichen HTML- und Skriptdateien. Die Checkliste kann dann wie folgt aufgerufen werden:

`http://localhost:43380/webapi/<Checkliste-Kurzname>/index.html`

Für den Fernzugriff ersetzen Sie „localhost“ durch die IP-Adresse des PCs, auf dem AAO läuft. Es gelten die gleichen Regeln wie für die WebAPI (= Bereitstellung von ein- und ausgehendem Zugriff über Ihre Firewall für Port 43380)

Importieren einer Textdatei

Der Dialog kann einfache Textdateien in einem vordefinierten Format importieren. Dies kann bei umfangreichen Checklistenprojekten mit vielen Abschnitten und Punkten hilfreich sein.

Das Importformat sieht wie folgt aus:

- Wenn eine Zeile in der Datei vollständig in Großbuchstaben geschrieben ist, wird sie als Beginn eines neuen Abschnitts angesehen
- „VOR DEM START“
„LANDE-CHECKLISTE“
usw.
- Wenn eine Zeile ein „Tabulatorzeichen“ enthält, wird sie als Frage-Antwort-Paar interpretiert
„Feststellbremse → EINGESCHALTET“
- Alle anderen Zeilen werden als Kommentare gespeichert und einfach auf der Webseite angezeigt. Kommentare werden nicht vorgelesen

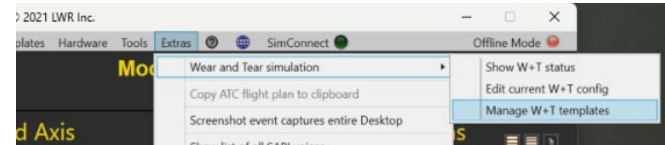
29 Verschleißsimulation

Das Verschleißmodul ist eine experimentelle Funktion in AAO. Es handelt sich im Wesentlichen um eine Hülle um die automatisierten RPN-Skripte, die dazu verwendet werden kann, die Alterung und den mechanischen Verschleiß von Flugzeugteilen sowie zufällige Ereignisse oder Ausfälle zu simulieren.

- Die W&T-Simulation ist in "**Templates**" (Vorlagen) organisiert.
- Jede Vorlage kann mehrere „**Systeme**“ enthalten.
- Jedes „**System**“ verfügt über eine Statusvariable, die den Verschleiß als Wert zwischen 0 % und 100 % anzeigt (= neu, 100 % = abgenutzt).
- Jedem „System“ können Sie „**Regeln**“ hinzufügen, die sich auf die Statusvariable auswirken. Eine „Regel“ kann ein einfacher Zähler sein, der jede Sekunde ein paar Punkte zum Verschleißprozentsatz hinzufügt, oder ein komplexeres Skript, das Punkte beispielsweise für Umgebungsbedingungen (Geschwindigkeit, Höhe, Temperatur usw.) hinzufügt.
- Schließlich kann jedes „System“ mehrere „**Konsequenzen**“ haben, die greifen, sobald der Verschleißstatus einen bestimmten Schwellenwert überschreitet. Dabei handelt es sich um Skripte, die entweder einmalig (und beispielsweise die Verbrennung in einem Motor zum Scheitern bringen) oder kontinuierlich (beispielsweise die Bremsen am linken Fahrwerk blockieren, um einen geplatzten Reifen zu simulieren) ausgeführt werden

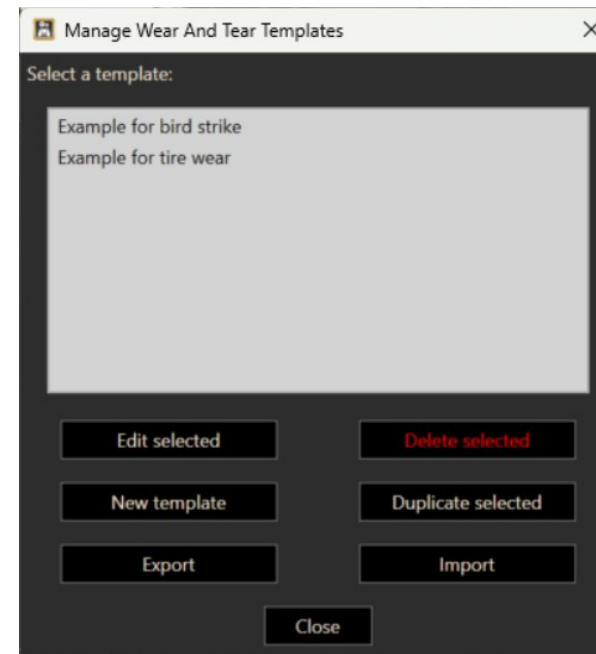
Wenn du eine Vorlage auf dein Flugzeug anwendest, erstellt die App eine Kopie der Vorlage für diese spezifische Flugzeuglackierung als „**Konfiguration**“ (derselbe Mechanismus wie bei den Schaltflächen und Achsen). Das bedeutet, dass eine Änderung der Vorlage nicht automatisch alle W&T-Konfigurationen für alle Flugzeuge ändert!

Die Funktion „Verschleiß“ befindet sich im Menü „Extras“



- „W+T-Status anzeigen“ öffnet ein Dialogfeld mit den aktuellen Werten der konfigurierten Systeme
- „Aktuelle W+T-Konfiguration bearbeiten“ ermöglicht es Ihnen, die Ihrem aktuellen Flugzeug zugewiesene Konfiguration zu ändern
- „W+T-Vorlagen verwalten“ dient zum Bearbeiten, Exportieren und Importieren von Vorlagen

Doppelklicken Sie auf eine Vorlage, um sie auszuwählen, oder verwenden Sie die Schaltflächen am unteren Rand des Dialogfelds



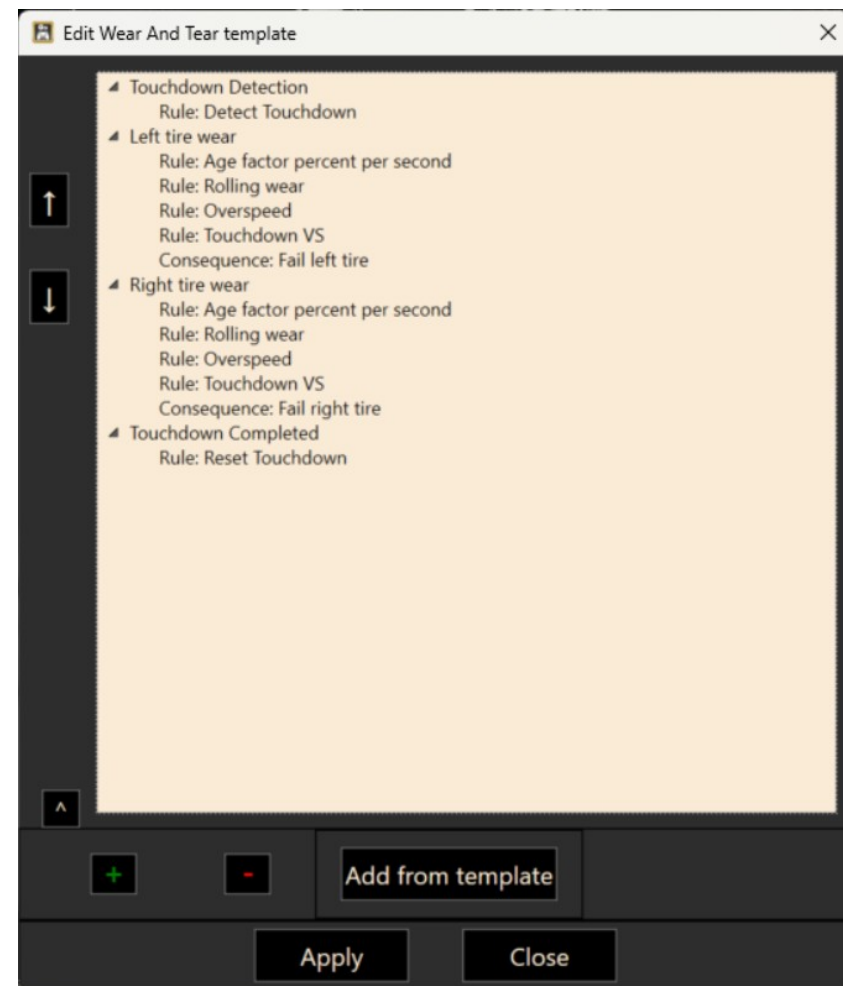
Eine Vorlage oder Konfiguration bearbeiten

Dieser Dialog sieht für Vorlagen und Konfigurationen gleich aus. Er listet die aktiven Systeme mit einer kurzen Beschreibung ihrer Regeln und Konsequenzen auf.

Mit „+“ können Sie ein neues System hinzufügen und mit „-“ das ausgewählte System entfernen.

Sie können auch Systeme aus einer anderen Vorlage hinzufügen.

Durch Doppelklicken auf ein System wird dieses im Editor-Dialog geöffnet.

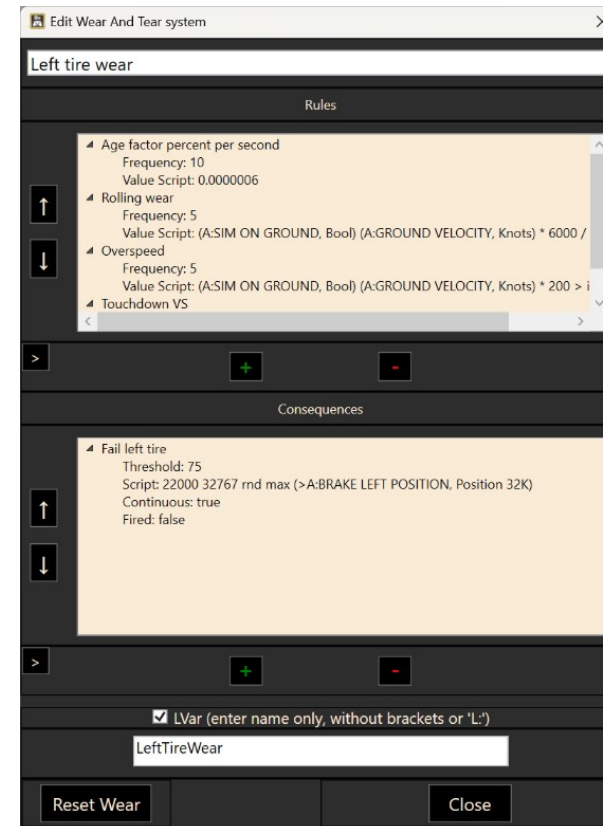


Ein System bearbeiten

In diesem Dialogfeld können Sie

- den Namen des Systems ändern
- Regeln hinzufügen, entfernen und neu anordnen
- Konsequenzen hinzufügen, entfernen und neu anordnen
- eine LVar Ihrer Wahl hinzufügen, um den aktuellen Verschleißprozentsatz dieses Systems zu speichern (falls Sie diesen auf Ihrer eigenen Anzeige oder Website anzeigen möchten)

Doppelklicken Sie auf eine Regel oder eine Konsequenz, um sie zu bearbeiten



Eine Regel bearbeiten

Regeln bestehen aus einem Wert-Skript und einem Aktualisierungs-Skript. Beide Elemente sind optional.

Der Zweck des Wert-Skripts besteht darin, einen Wert bereitzustellen, der den Verschleißprozentsatz des Systems bei jeder Auswertung der Regel erhöht.

Das Aktualisierungsskript dient für zusätzlichen Code, der bei jeder Aktualisierung der Regel ausgeführt werden soll.

Die Häufigkeit der Aktualisierungen kann mit dem Mausrad eingestellt werden.

Sie können auf das Alter des Systems zugreifen, indem Sie „(L:sysage)“ in Ihrem Code verwenden. Dies ist keine echte LVar; der Literal wird zur Laufzeit durch das tatsächliche Alter dieses Systems in Stunden seit dem letzten Reset ersetzt.

Header: Rolling wear

Value Script: (A:SIM ON GROUND, Bool) (A:GROUND VELOCITY, Knots) * 6000 /

Update Script

Frequency in 100ms: 5 1 = check/update 10 times per second, 10 = once per second, etc.

Buttons: Cancel, Apply

Eine Konsequenz bearbeiten

Eine Konsequenz enthält ein Skript, das ausgeführt wird, sobald der Verschleißprozentsatz des Systems den ausgewählten Schwellenwert überschreitet.

Normalerweise wird das Skript nur einmal ausgeführt, aber Sie können „Kontinuierlich“ auswählen, damit es ständig ausgelöst wird.

In diesem Beispiel wird die linke Bremse ständig mit einer zufälligen Kraft betätigt, wodurch ein geplatzter Reifen simuliert wird.

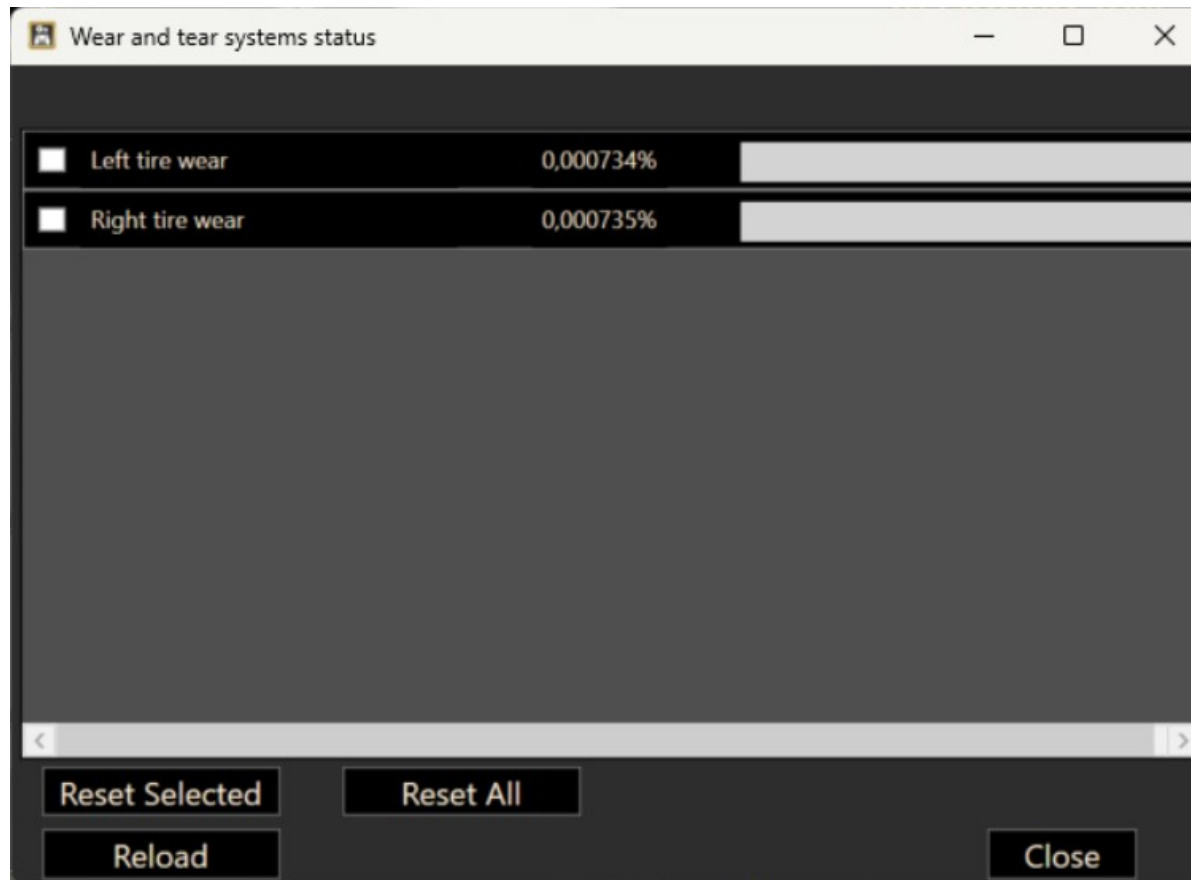
The screenshot shows a dialog box titled "Edit Consequence". The dialog is divided into four sections by dashed lines:

- Header:** Contains the text "Fail left tire".
- Script:** Contains the code "22000 32767 rnd max (>A:BRAKE LEFT POSITION, Position 32K)".
- Threshold in %:** Contains a text input field with the value "75".
- Continuous:** Contains a checked checkbox.

At the bottom of the dialog, there are two buttons: "Cancel" and "Apply".

Verschleißstatus

Wenn für Ihr Flugzeug eine Verschleißkonfiguration angelegt wurde, können Sie den aktuellen Verschleißprozentsatz abrufen. Sie können den Status auch auf 0 % zurücksetzen, um das System zu „reparieren“.



30 Zuordnung von LVar zu BVar (veraltet, bitte nicht verwenden)

(BVars werden nun wie alle anderen Variablen behandelt – eine Zuordnung ist nicht mehr erforderlich)

Mit der LVar/BVar-Zuordnungslogik ist es möglich, zusätzlichen Code in die MSFS-Animationsverarbeitungs-Pipeline einzufügen. Diese Zuordnungslogik steht weiterhin als Alternative zum neuen integrierten BVar-Zugriff zur Verfügung, falls dieses Modul ausfallen sollte.

Die Grundidee besteht darin, frei wählbare LVars an die BVar-Eingabeereignisse in MSFS anzuhängen, sodass Sie das BVar-Ereignis durch Ändern des Werts der LVar auslösen können.

Dies muss mit RPN-Skripten wie diesem erfolgen:

```
1 · (L:AAO_JU52_HDG_HOLD, ·Bool) ·- · (>L:AAO_JU52_HDG_HOLD, ·Bool)
```

Die LVar „(L:AAO_JU52_HDG_HOLD, Bool)“ wird auf den Wert 0 oder 1 gesetzt und gibt diesen Wert an die zugehörige BVar „(B:JU52_AUTOPILOT_Heading_Hold, Bool)“ weiter (siehe folgende Seite). AAO sorgt dafür, dass das Eingabeereignis nicht ständig ausgelöst wird. Aufgrund dieses Vorgangs können Sie Schalter nicht schneller als etwa 10 Klicks pro Sekunde betätigen!

Alle Konfigurationen können in einzelne „Profile“ gepackt werden (zum Beispiel eines pro Flugzeug).

!Die Profile müssen aktiviert und der Hook muss aktiviert sein, BEVOR SIE MSFS STARTEN!

Im Menü „Extras“ können Sie über „Map BVars to LVars (MSFS)“ auf den Hook-Konfigurationsdialog zugreifen.

Manage LVar to BVar hook

Select profile: First profile Delete

Profile name: First profile Add as new Rename

```
(L:AAO_JU52_HDG_HOLD, 'Bool) · (B:JU52_AUTOPILOT_Heading_Hold, 'Bool) != ·if{ ·1 · (>B:JU52_AUTOPILOT_Heading_Hold_Toggle) ·} ·
(L:AAO_TBM_FUEL_Selection_Auto, 'Number) · (B:FUEL_Selection, 'Number) != ·if{ ·(L:AAO_TBM_FUEL_Selection_Auto, 'Number) · (>B:F
(L:AAO_CJ4_flood_set, 'Number) · (B:LIGHTING_CABIN_3, 'Number) != ·if{ ·(L:AAO_CJ4_flood_set, 'Number) · (>B:LIGHTING_CABIN_3_Se
(L:AAO_DEICE_Windshield_1, 'Bool) · (B:DEICE_Windshield_1, 'Bool) != ·if{ ·(>B:DEICE_Windshield_1_toggle) ·} ·
(L:AAO_TBM_InertSep, 'Bool) · (B:DEICE_Engine_1, 'Bool) != ·if{ ·1 · (>B:DEICE_Engine_1_Toggle) ·} ·
```

Edit selected: Clear fields Scan MSFS for BVars Delete selected

LVar: (L:AAO_TBM_InertSep, Bool)

BVar get: (B:DEICE_Engine_1, Bool)

BVar set: 1 (>B:DEICE_Engine_1_Toggle)

Initialize LVar with BVar get
 Sync LVar with BVar get

Overwrite selected

Export profile Import profile

Close Disable hook Enable hook Activate profiles

Profile hinzufügen, löschen oder umbenennen

Aufträge im ausgewählten Profil

Doppelklicken Sie auf einen Auftrag, um ihn in die Bearbeitungsfelder zu laden

Geben Sie hier die Zuweisungseigenschaften ein:

LVar: Sie können einen beliebigen LVar-Namen wählen und festlegen, ob Sie ihn mit der aktuellen Schalterposition initialisieren oder ständig synchronisieren möchten (damit Sie die Bedienelemente sowohl extern als auch im VC bedienen können)

BVar get: Der aus MSFS zu lesende BVar-Wert

BVar set: Das auszulösende Eingabeereignis

So finden Sie Informationen zu BVar-Ereignissen

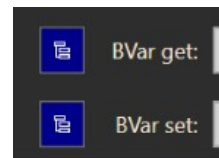
In den meisten Fällen gilt: Wenn ein Schalter oder Regler im Cockpit mit keiner der klassischen Methoden (Sim-Ereignisse, LVars oder HVars) funktioniert, ist die Wahrscheinlichkeit groß, dass er über eine BVar gesteuert wird. Bitte lesen Sie die MSFS-SDK-Dokumentation zu den Einzelheiten der neuen „INPUT EVENTS“.

Den Namen einer bestimmten BVar finden Sie entweder in der Model-Behavior-Definitionsdatei des Flugzeugs oder über die ModelBehavior-Debug-Konsole im MSFS-Entwicklermodus.

Mit der Schaltfläche „**Scan MSFS for BVars**“ (MSFS nach BVars durchsuchen) können Sie AAO nach den BVars suchen lassen, die im Simulator direkt zugänglich sind.

Diese Methode findet keine BVars, die Teil eines „Premium“-Flugzeugs mit verschlüsselten Konfigurationsdateien sind.

Sobald die Variablen erfasst wurden, können Sie die beiden kleinen Baumlisten-Schaltflächen neben „BVar get“ und „BVar set“ verwenden, um aus einer Baumliste geeignete Kandidaten für Ihren Code auszuwählen.



Wie bei den RPN-Skripten können Tags (#...#) in einem Variablennamen durch Doppelklick auf den Tag-Namen in der Baumliste ersetzt werden. Wählen Sie zunächst die Variable mit den Tags durch Doppelklick aus, scrollen Sie dann in demselben Dialog durch die Tag-Liste und doppelklicken Sie auf den gewünschten Wert. Dadurch wird der Tag im Eingabefeld durch den tatsächlichen Wert ersetzt.

BVars im RPN-Code

BVar-Eingabeereignisse werden nicht wie LVars und andere Variablentypen durch ein einzelnes Literal dargestellt.

- Zunächst gibt es den „Getter“-Teil, also den Variablennamen ohne angehängte „_set, _toggle, push“ usw. Dieser gibt den aktuellen Zustand der Eingabeereignisvariablen zurück (0, 1, 2 usw.).
Beispiel: (B:Fuel_Selection, Bool)
- Zweitens gibt es mehrere „Setter“, wie „_Set“, „_Toggle“, „_Push“, „_Inc“, „_Dec“, „_Left“, „_Right“, „_Up“, „_Down“. Diese werden einfach an den Namen des BVar-Getters angehängt, und Sie können ihnen Werte mit (>B:varname_setter) zuweisen.
Beispiele: 1 (>B:Fuel_Selection_Set, Bool), 0 (>B:Fuel_Selection_Set, Bool), 1 (>B:Fuel_Selection_Toggle, Bool) usw.

Wie genau Sie das Eingabeereignis auslösen müssen, hängt davon ab, was der Simulator als Setter erwartet. Sie müssen also den vollständigen Setter-Code in das letzte Feld eingeben.

Ein komplexeres Beispiel, das die BVar effektiv mit Ihrer LVar synchronisiert:

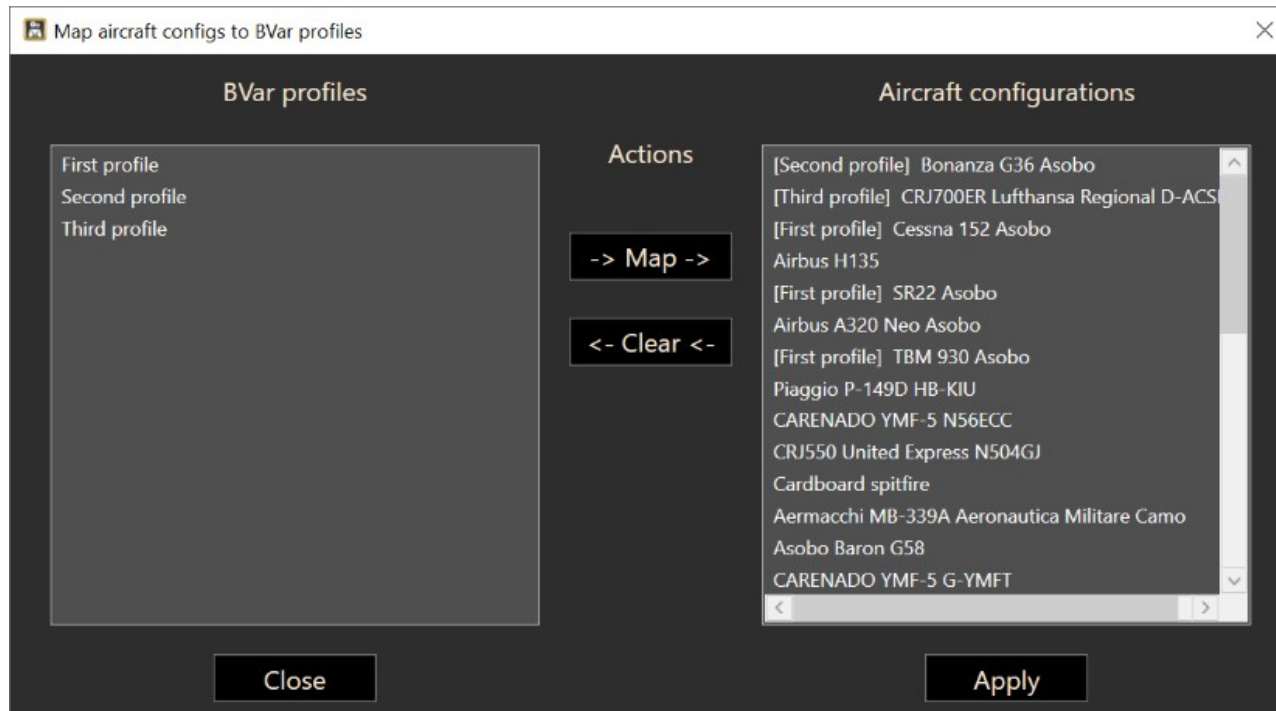
LVar: (L:AAO_TBM_FUEL_Selection_Auto, Number)

BVar get: (B:FUEL_Selection, Number)

BVar set: (L:AAO_TBM_FUEL_Selection_Auto, Number) (>B:FUEL_Selection_Set)

Zuweisen der BVar-Profile zu Ihrem Flugzeug

Um die in Ihren Profilen definierten LVars nutzen zu können, müssen Sie das jeweilige Profil einer bestehenden Flugzeugkonfiguration zuweisen. Verwenden Sie den Dialog „Extras → Map BVar profiles to aircraft“ (Extras → BVar-Profile Flugzeugen zuweisen), um Ihre Profile zuzuweisen und zu aktivieren.



So weisen Sie ein Profil zu: Wählen Sie mit der Maus ein Profil und anschließend ein Flugzeug aus und klicken Sie dann auf „Map“ (Zuordnen).

So entfernen Sie ein Profil aus einem Flugzeug: Wählen Sie das Flugzeug aus und klicken Sie dann auf „Clear“ (Löschen).

Klicken Sie auf „Apply“ (Übernehmen), um Ihre Änderungen zu speichern.

Profile gemeinsam nutzen

Über die Schaltfläche „Profil exportieren“ kannst du ein Profil in eine XML-Datei speichern und diese XML-Dateien über die Schaltfläche „Profil importieren“ laden. Auf diese Weise kannst du deine Profile mit anderen Simmern teilen.

Ein Profil aktivieren

Bevor du die Simulation startest, öffne das Dialogfeld, wähle ein Profil aus der Dropdown-Liste aus und klicke auf die Schaltfläche „Profil aktivieren“.
Wenn die Simulation bereits läuft, hat dies keine Wirkung – Sie müssen sie neu starten!

Aktivieren und Deaktivieren des „Hooks“ (einhaken)

Die Zuordnung funktioniert nur, wenn die Hook-Logik aktiviert ist. Da dieser Hook ziemlich invasiv ist, haben Sie die Wahl, ihn zu aktivieren oder zu deaktivieren. Standardmäßig ist der Hook deaktiviert.

Was tun, wenn die BVar-Zuordnung bei einem bestimmten Flugzeug nicht funktioniert?

Wie bei allem in MSFS gibt es Möglichkeiten, den AAO-BVar-Hook zu umgehen. Das sollte eigentlich nicht vorkommen, tut es aber doch. Wenn Sie ein Flugzeug finden, bei dem die BVar-Zuordnung nicht zu funktionieren scheint, können Sie den AAO-Hook manuell hinzufügen.

Um den BVar-Hook manuell einzufügen, müssen Sie

1. sicherstellen, dass der Simulator nicht läuft
2. die Modellverhaltens-Definitionsdatei des Flugzeugs suchen. Sie befindet sich normalerweise im Ordner „model“ des Flugzeugpakets und heißt etwa „planename_Interior.XML“ oder ähnlich
3. die Datei mit einem erweiterten Editor wie Notepad++ öffnen
4. nach unten scrollen, bis Sie das erste „<Component ID=“...“-Tag sehen
5. den folgenden Code VOR diesem ersten „<Component“ einfügen:
<Include ModelBehaviorFile="LORBY_Lvar_To_Bvar.xml"/>
<Component ID="LORBY">
<UseTemplate Name="AAO_Lvar_To_Bvar_Handler" />
</Component>
6. die Datei speichern und den Simulator neu starten

```
<?xml version="1.0" encoding="utf-8" ?>
<ModelInfo>
  <LODS>
    <!-- Highest LOD -->
    <!-- minimum display size in % (should be bounding sphere vertical screen s
    <!-- default minSize is 0 -->
    <LOD_minSize="150" ModelFile="A320_NEO_INTERIOR_LOD00.gltf"/>
    <LOD_minSize="80" ModelFile="A320_NEO_INTERIOR_LOD01.gltf"/>
    <LOD_minSize="50" ModelFile="A320_NEO_INTERIOR_LOD02.gltf"/>
    <LOD_minSize="15" ModelFile="A320_NEO_INTERIOR_LOD03.gltf"/>
    <LOD_minSize="5" ModelFile="A320_NEO_INTERIOR_LOD04.gltf"/>
    <LOD_minSize="1" ModelFile="A320_NEO_INTERIOR_LOD05.gltf"/>
    <LOD_minSize="0.1" ModelFile="A320_NEO_INTERIOR_LOD06.gltf"/>
  </LODS>
  <Behaviors>
    <Include ModelBehaviorFile="Asobo\Common.xml"/>
    <Include ModelBehaviorFile="Asobo\Airliner\AirlinerCommon.xml"/>
    <Include ModelBehaviorFile="Asobo\Airliner\FMC.xml"/>
    <Include ModelBehaviorFile="Asobo\Airliner\GlassCockpit.xml"/>
    <Include ModelBehaviorFile="Asobo\Airliner\Airbus.xml"/>
    <Include ModelBehaviorFile="LORBY_Lvar_To_Bvar.xml"/>
  </Behaviors>
  <Component ID="LORBY">
    <UseTemplate Name="AAO_Lvar_To_Bvar_Handler" />
  </Component>
  <!-- RETROECLAIRAGE #####
  <Component ID="MCDU">
    <DefaultTemplateParameters>
      <LONG_CLICK_TIME>1</LONG_CLICK_TIME>
    </DefaultTemplateParameters>
    <UseTemplate Name="ASOBO_FMC_A320">
      <ID>1</ID>
      <POTENTIOMETER>15</POTENTIOMETER>
      <NODE ID DTR>PUSH MCDUL DIR</NODE ID DTR>
```

31 vJoy-Schnittstelle

Wenn Sie vJoy auf Ihrem Computer installiert haben, können Sie mit AAO virtuelle Joystick-Tastendrücke oder Achsbewegungen an eine Anwendung senden.

Damit ist es beispielsweise möglich, Funktionen auszulösen, die in der SimConnect-API nicht verfügbar sind, indem Sie eine vJoy-Taste im Simulator selbst zuweisen und diese dann über AAO auslösen.

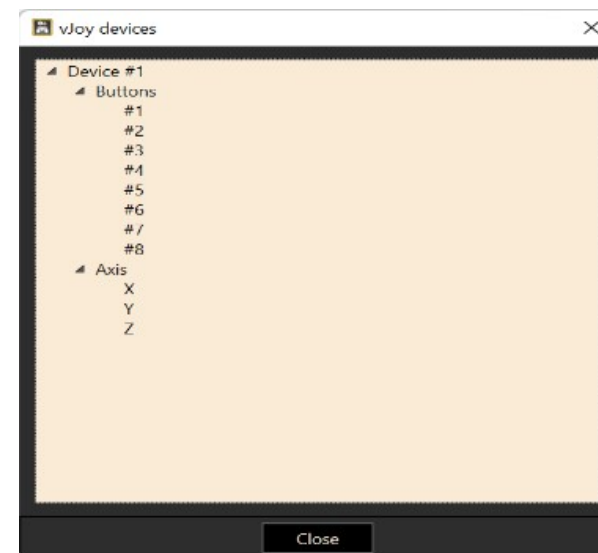
Darüber hinaus kann AAO im Offline-Modus als „Brücke“ zwischen ausgefallenerer Hardware und jeder App fungieren, die mit Joystick-Tasten oder -Bewegungen gesteuert werden kann. Die Verwendung von AAO ist also nicht mehr auf Flugsimulationen beschränkt. Wenn Sie beispielsweise ein Behringer X-Touch mit Ihrem Lieblings-Rennspiel verwenden möchten – das ist möglich.

vJoy kann von der Projekt-Website <https://sourceforge.net/projects/vjoystick/> heruntergeladen werden.

Die Schnittstelle kann im AAO-Menü unter **„Hardware->vJoy-Schnittstelle aktivieren“** aktiviert werden.

Alle vJoy-Geräte, die für AAO verfügbar sind, werden unter **„Hardware->vJoy-Geräte anzeigen“** angezeigt.

vJoy erlaubt es nur einer App gleichzeitig, ein virtuelles Gerät zu steuern. Wenn ein Gerät bereits in Gebrauch ist, wird es in AAO nicht angezeigt.



32 ViGEm-Schnittstelle

ViGEm ist eine kleine Software, die einen Xbox 360-Controller emuliert (ähnlich wie vJoy es für Joysticks tut).

ViGEm kann von der Projekt-Website <https://github.com/ViGEm/ViGEmBus/releases> heruntergeladen werden.

Die Schnittstelle kann im AAO-Menü unter
„Hardware->ViGEm-Schnittstelle aktiviert“
aktiviert werden.

Dies ist nützlich, um jede Hardware, die AAO verarbeiten kann, mit Anwendungen zu verbinden, die nur Xbox-Controller akzeptieren.

33 Virtuelle Maus

Mit AxisAndOhs kannst du Ereignisse an deine Maus senden. Das bedeutet, dass du die Maus mit einem externen Gerät steuern, den Cursor auf dem Bildschirm positionieren sowie Mausklick- und Mausevent-Ereignisse auslösen kannst.

Virtuelle Mausevente können sowohl Achsen als auch Tasten zugewiesen werden.

Virtuelle Maustasten und Aktionen:

Wenn Sie in AAO eine Taste oder eine Aktion zuweisen, können Sie Mausaktionen als „virtuelle Ereignisse“ auswählen.

Die Aktionen „Bewegen“ und „Rad“ können über die Option „Bewegungsskala“ skaliert werden. Höhere Werte bedeuten eine größere Bewegung des Mauszeigers auf dem Bildschirm.

Virtuelle Mausachse:

Wenn du eine Achse zuweist, kannst du wählen, ob du die X- oder Y-Mausbewegung an diese Achse binden möchtest. Damit dies funktioniert, musst du auch eine „Combo“-Taste für die Achse konfigurieren, da du dich sonst komplett aus der Maussteuerung aussperren könntest. Für eine Mausachse musst du „Axis Min“ und „Axis Max“ so anpassen, dass sie zu deiner Bildschirmauflösung passen; die Mausposition ist absolut. Für relative Bewegungen verwende die Mausaktionen „Move...“ im Trigger-Modus der Achse.

34 CAN-Schnittstelle

AxisAndOhs kann CAN-Nachrichten (Controller Area Network) über zwei Arten von physikalischen Schnittstellen lesen:

- „CANable“-USB-Schnittstelle mit „Candle“-kompatibler Firmware
- CAN-USB-HID-Schnittstelle

CAN-Geräte

Die Schnittstelle wurde erfolgreich mit folgenden Geräten getestet:

1. Arduino UNO mit „keyestudio.com“ CAN-BUS-Shield
2. Arduino Mega mit „seeedstudio.com“ CAN-Bus-Shield V2
3. CAN-HID-Hardware von CIS
[CAN-Bus-Integration für Flugsimulatoren MSFS FSX X-Plane P3D \(caninsimulation.de\)](http://caninsimulation.de)

Senden von CAN-Nachrichten an Hardwaregeräte mit „device-id“:

AAO Befehl	Parameter	Inhalt
(CANMSG:device_id can_id data)	'can_id' (2 byte hex, 'hhhh')	Die Daten bestehen aus einer Reihe von Bytes im zweistelligen Hexadezimalformat
(CANMSG:device_id can_id header n f)	'can_id' (2 byte hex, 'hhhh')	n ist eine Gleitkommazahl
(CANMSG:device_id can_id header n i)	'can_id' (2 byte hex, 'hhhh')	n ist eine ganze Zahl (Integer)

CAN-Nachrichtenprotokoll

AAO verarbeitet die folgenden Nachrichten:

1. CIS-Standard (CAN in der Simulation)

CIS ist eine Teilmenge des CANas-Standards (CAN in der Luft- und Raumfahrt). Er deckt die typischen Eingabeereignisse ab, die von Simulationssteuerungen zu erwarten sind, wie Joysticks, Steuerknüppel, Drehgeber, Tastaturen und Schaltkästen.

Nachrichtenformat (8 Byte)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Node ID	Data type	Encoder ID	Message counter	m1	m2	m3	m4

Node ID (Knoten-ID): Eindeutige numerische ID des CAN-Knotens, der die Nachricht sendet

Data type (Datentyp): Kennung gemäß CANas-Standard

Encoder ID (Encoder-ID): ID des Encoders, der die Daten sendet. Jeder Knoten kann 256 Encoder jedes Typs (Drehgeber, Schalter, Achse, Tastatur) haben

Message Counter (Nachrichtenzähler): 0–255 (erhöht sich mit jeder neuen Nachricht)

m1–m4: Datenbytes

Nachrichtentypen

Typ	CAN-ID-Bereich	Datentyp	Datenbytes
Rotary Encoder (Rotationsgeber)	708h - 70Fh	0Bh (BCHAR)	m1: Bit 7: fast rotation (Schnelldrehung) Bit 3: Event PUSHBUTTON OFF Bit 2: Event PUSHBUTTON ON Bit 1: Event DOWN Bit 0: Event UP
Switch (Schalter)	710h - 717h	0Bh (BCHAR)	m1: Bit 1: Event OFF Bit 0: Event ON
Analoge Achse (Analog (Axis))	718h - 71Fh	07h (USHORT)	m1, m2 16 bit integer value (0-65535)
Keyboard (Tastatur)	720h - 727h	13h (UCHAR2)	m1: modifier code (Modifikatorcode) m2: key code (Tastencode)
Wert (ergibt eine LVar)	730h - 73Fh	02h, 03h, 04h, 05h, 19h	m1, m2 16 bit integer value

2. CANaerospace-Standard, beschränkt auf die Achsen-Eingangsnachrichten im CAN-ID-Bereich 4xx gemäß dem Kapitel „Flight controls data“ in der Spezifikation „canas_17.pdf“ von „Stock Flight Systems“.

35 Soundeffekte für Audio- und Sprachausgabe

Mit den Befehlen SOUND und SPEAK können Sie eine Vielzahl von Soundeffekten verwenden. Soundeffekte werden wie folgt zu den Skriptbefehlen hinzugefügt:

- **Beim Abspielen von Audiodateien** werden die Effekte am Ende des Befehls hinzugefügt

(SOUND:xxxx|vvv|bbb|dd|eee-fff-ggg)

Dadurch wird die Audiodatei xxxx mit der Lautstärke vvv und der Balance bbb auf dem Gerät mit der ID dd unter Verwendung der Effektkette eee-fff-ggg abgespielt

- **Bei der Text-to-Speech-Ausgabe** wurden die Soundeffekte in einen separaten Befehl verschoben

(VOICEEFFECTS:eee-fff-ggg)

Beachten Sie, dass VOICE-Befehle persistent sind. Sie müssen die Effekte deaktivieren, wenn Sie sie bei Ihrem nächsten SPEAK-Befehl nicht verwenden möchten, indem Sie (VOICEEFFECTS:0) verwenden.

Format der Effektreihenfolge

Eine Effektkette wird erstellt, indem die Effekt-ID, die Parameter und die Lautstärke durch Kommas getrennt angegeben und weitere Effekte durch ein Minuszeichen getrennt hinzugefügt werden. Die Effekte bauen aufeinander auf, daher ist die Reihenfolge, in der Sie sie der Reihe hinzufügen, wichtig.

(VOICEEFFECTS:7,880,20-1,3-4,50,50)

Diese Effektkette enthält drei Effekte:

- einen Sinuston mit 880 Hz
- ein weißes Rauschen bei geringer Lautstärke (3)
- einen mittleren Verzerrungseffekt (50) bei mittlerer Lautstärke (50)

Wenn der nächste SPEAK-Befehl aufgerufen wird, hörst du ein leises Brummen, etwas weißes Rauschen und die verzerrte Stimme.

(SOUND:mywave.mp3|100|0|-1|7,880,20-1,3-4,50,50): Spiel die Audiodatei mit Lautstärke 100 und mittlerer Balance auf dem Standard-Audiogerät mit der Effektkette ab

Effekt	ID	Parameter	Beispiel
white noise (weißes Rauschen)	1	volume (Lautstärke)	1,3 - low hissing sound (leises Rauschen)
pink noise (rosa Rauschen)	2	volume (Lautstärke)	2,50 - strong noise (starkes Rauschen)
flanger (Flügelfilter)	3	volume (Lautstärke)	3,50 - moderate Flanger effect (mäßiger Flanger-Effekt)
Distortion (Verzerrung)	4	amount,volume (Anzahl,Lautstärke)	4,10,50 - adds crackles and distortion (fügt Knackgeräusche und Verzerrungen hinzu)
Delay (Delay)	5	amount,volume (Anzahl,Lautstärke)	5,100,50 - strong echo effect (starker Echoeffekt)
tremolo	6	amount,volume (Anzahl,Lautstärke)	6,5,100 - makes the sound wobble (lässt den Ton schwanken)
sine wave (Sinuswelle)	7	frequency,volume (Frequenz,Lautstärke)	7,440,25 - low standard pitch hum (leises Brummen mit Standardtonhöhe)
square wave (Rechteckwelle)	8	frequency,volume (Frequenz,Lautstärke)	8,880,25 - high vibrating hum (hohes vibrierendes Brummen)
sawtooth wave (Sägezahnwelle)	9	frequency,volume (Frequenz,Lautstärke)	9,1320,25 - higher pitched vibrating hum (höheres vibrierendes Brummen)

**Alle Parameterwerte liegen zwischen 0 und 100, mit Ausnahme der Frequenz, die in Hz angegeben wird.
Soundeffekte einzeln abspielen**

Soundeffekte können auch separat, für sich allein, abgespielt werden:

(PLAYEFFECT:id|vvv|bbb|dd|eee-fff-ggg|ms)

spielt die Effektkette **eee-fff-ggg** auf dem Gerät **dd** mit der Lautstärke **vvv** und der Balance **bbb** für **ms** Millisekunden ab.
Wenn ms auf 0 gesetzt wird, wird die Effektkette unbegrenzt abgespielt.

Die „**id**“ ist eine Zeichenfolge Ihrer Wahl, um das Signal zu identifizieren, wenn Sie es anhalten möchten:

(STOPEFFECT:id)

36 ChatGPT-Schnittstelle

Wenn Sie über ein kostenpflichtiges ChatGPT-API-Konto bei OpenAI verfügen, können Sie AAO nutzen, um Textantworten von der KI zu erhalten.

Hier können Sie sich für die OpenAI-API anmelden:

[Produkt \(openai.com\)](https://openai.com) Klicken Sie auf „Get started“

Nachdem Sie Ihr Konto erstellt haben, klicken Sie auf das „Blumen“-Symbol in der oberen linken Ecke und wählen Sie „API keys“. Dort können Sie einen neuen Schlüssel erstellen. Lesen Sie unbedingt die Preisinformationen, denn die ChatGPT-API ist kein kostenloser Dienst.

Geben Sie anschließend Ihren API-Schlüssel unter „Extras -> ChatGPT-API-Schlüssel eingeben“ ein. Dies ist nicht dasselbe wie Ihre Anmeldedaten für die ChatGPT-App; Sie müssen ein eigenes API-Konto erstellen. Beachten Sie, dass dies nicht kostenlos ist; die Nutzung der API ist kostenpflichtig. Derzeit können Sie mit etwa 1 Cent pro Anfrage rechnen, je nach Umfang der Anfrage.

Wenn Sie Ihren Schlüssel eingegeben haben, können Sie ChatGPT mit RPN-Befehlen abfragen. AAO simuliert ein „Chat-Abschluss“-Szenario, bei dem Sie der KI mitteilen, wer sie ist und wie die Situation aussieht, und sie dann um eine Antwort bitten.

(CHATGPTADDMESSAGE:role:text) fügt dem Szenario einen Eintrag hinzu. Beginne mit der Rolle „system“, um die Situation zu beschreiben, und verwende dann „user“ und „assistant“, um den Chat aufzubauen.

(CHATGPTGETRESULT:engine|lvarname) sendet die Anfrage an die KI. „engine“ kann entweder „gpt-3.5-turbo“ oder „gpt-4“ sein. Der resultierende Text wird in „(L:lvarname, String)“ geschrieben.

Beispielscript:

```
' ' (>L:purserspeak, ·String) ·\n
(ChatGPTADDMESSAGE:system:you ·are ·the ·purser ·on ·the ·lufthansa ·flight ·DLH2345 ·from ·munich ·to ·rome. ·It ·
is ·raining ·and ·the ·temperature ·is ·5 ·degrees ·celsius. ·The ·flight ·is ·departing ·at ·20:00) ·\n
(ChatGPTADDMESSAGE:user:please ·read ·the ·cabin ·announcement) ·\n
(ChatGPTGETRESULT:gpt-3.5-turbo|purserspeak) ·\n
```

Ergebnis:

(L:purserspeak, String)

enthält

"Ladies and gentlemen, this is your purser speaking. On behalf of Lufthansa, I would like to welcome you aboard flight LH2345 from Munich to Rome.

I would like to inform you that our departure is scheduled for 20:00. However, due to the inclement weather conditions, it is currently raining with a temperature of 5 degrees Celsius in Munich. We kindly ask for your patience as we prepare the aircraft for departure.

As always, your safety and comfort are our top priority. Please ensure that your seatbelts are fastened securely, your electronic devices are switched off or in airplane mode, and your tray tables and seatbacks are in their upright position.

During the flight, our cabin crew will be available to assist you with any needs you may have. We offer a range of complimentary beverages and light snacks for your enjoyment during the flight.

Once again, we apologize for any inconvenience caused by the weather conditions. We will make every effort to minimize any delays and provide you with a pleasant flight experience.

Thank you for choosing Lufthansa, and we hope you have a wonderful journey to Rome."

Die ChatGPT-API ist ziemlich langsam. Wenn Sie beispielsweise vorhaben, sich das Ergebnis vorlesen zu lassen, müssen Sie eine SPLIT- oder GOTO-Logik einbauen und auf das Ergebnis warten:

```
'-' ·(>L:purserspeak, ·String) ·\n
(ChatGPTADDMESSAGE:system:you ·are ·the ·purser ·on ·the ·lufthansa ·flight ·DLH2345 ·from ·munich ·to ·rome. ·It ·
is ·raining ·and ·the ·temperature ·is ·5 ·degrees ·celsius. ·The ·flight ·is ·departing ·at ·20:00) ·\n
(ChatGPTADDMESSAGE:user:please ·read ·the ·safety ·instructions) ·\n
(ChatGPTGETRESULT:gpt-3.5-turbo|purserspeak) ·\n
:waitcomplete ·\n
(L:purserspeak, ·String) ·slen ·1 ·== ·if{ ·(GOTO:waitcomplete) ·} ·\n
(VOICE:Azure|de-DE-ElkeNeural) ·(VOICEEFFECTS:4,100) ·(L:purserspeak, ·String) ·(SPEAK:%s1) ·\n
```

37 Befehlszeilenparameter

Die EXE-Datei von AxisAndOhs kann mit den folgenden Befehlszeilenparametern aufgerufen werden

```
-blacklistall
```

Dadurch werden alle Geräte, die AAO erkennt, auf die Blacklist gesetzt. Verwenden Sie diesen Parameter, wenn AAO aufgrund von USB-Hardwareproblemen nicht starten will.

```
-offline „aircraft livery“
```

Startet AAO im Offline-Modus und lädt die Konfiguration für das Flugzeug.

38 Haftungsausschluss

Lorby AxisAndOhs wird ausschließlich für den privaten Gebrauch lizenziert, nicht verkauft. Alle Eigentumsrechte verbleiben beim Autor. Es ist Ihnen untersagt, dieses Paket oder Teile davon zu verbreiten. Das Zerlegen, Umgestalten oder Vornahme von Änderungen jeglicher Art ist untersagt.

Haftungsausschluss. Der Autor lehnt im gesetzlich zulässigen Umfang jegliche ausdrückliche oder stillschweigende Gewährleistung ab, einschließlich, aber nicht beschränkt auf stillschweigende Gewährleistungen hinsichtlich Rechtsmängeln, Nichtverletzung von Rechten Dritter, Marktgängigkeit oder Eignung für einen bestimmten Zweck. Ohne Einschränkung des Vorstehenden übernimmt der Autor ausdrücklich keine Gewähr dafür, dass:

- die Software Ihren Anforderungen oder Erwartungen entspricht;
- die Software oder der Softwareinhalt frei von Fehlern, Viren oder anderen Mängeln ist;
- alle durch die Software bereitgestellten oder generierten Ergebnisse, Ausgaben oder Daten korrekt, aktuell, vollständig oder zuverlässig sind;
- die Software mit Software von Drittanbietern kompatibel ist;
- Fehler in der Software behoben werden oder dass eine Weiterentwicklung stattfindet;
- die Software keine Fehler oder Schäden an dem Computersystem verursacht, auf dem sie installiert ist.

© 2021 Lorby Wildfire Response Inc.
Kamloops BC, Kanada

support@wildfiretrainingsolutions.ca

<https://www.avsim.com/forums/forum/788-lorby-si-support-forum/>

<https://www.wildfiretrainingsolutions.ca/>