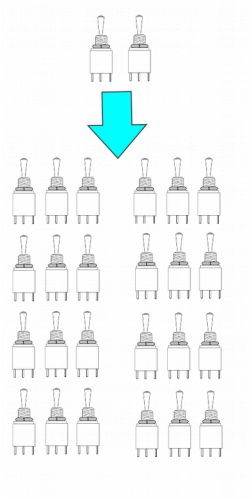
Anleitung zum source-script „Virtual Switch“



Inhaltsverzeichnis

Einleitung....2
Allgemeines....2
Was lässt sich Auslösen / Schalten ?....2
Grundidee:....2
Hinweise vorab:....3
Funktionsweise:....4
Bedienung....4
Installation:....5
(1) Dateien & Ordner kopieren....5
(2) source script einrichten....5
(3, optional) Logische Schalter einrichten....6
Konfiguration:....7
Einleitung:....7
Konfigtabelle pflegen....7
Liste möglicher Funktionen....8
Sounds....9
Sonstige Optionen....10
Neigungs Schwellwerte....10
Tastzeiten....10
Akustik-feedback....10

Einleitung

Allgemeines

VirtualSwitch schafft es mit nur zwei Tastern bis zu **24 Schaltfunktionen** auszulösen.

Dies wird durch Unterscheidung in der „Tastzeit“ (also kurzes, mittleres oder langes Drücken) in Kombination mit den evtl in Sendern integrierten Gyros erzielt.

Über die Gyros wird die Senderhaltung (normal, links, rechts, nach vorne) ermittelt.

Sender ohne Gyro können dementsprechend nur sechs Schaltfunktionen abbilden.

Was lässt sich Auslösen / Schalten ?

Aus sicherheitsrelevanten Gründen sollten für die eigentliche Steuerung von Modellen nur nebensächliche Funktionen ausgelöst werden.

z.B.:
- Ansage von Telemetriewerten
- Ansage von Timern
- Resetten von Timern, einzelne Telemetriesensoren (Verbrauch , Höhe…) oder gesamte Telemetrie
- Umschalten („toggeln“) von maximal zwei Logischen Schaltern

Grundidee:

Zum einen ermöglicht die Verwendung der im Boden des Sendergehäuses angebrachten Tastern das Auslösen vieler Funktionen ohne die Finger vom Knüppel zu nehmen.
Außerdem „entlastet“ es ggf Schalter & Trimmer, die nun für andere wichtige Funktionen zur Verfügung stehen.

Typischerweise besitzt man von einer Klasse an Modellen (Zwecksegler, Scale Segler, Motormodelle..) mehrere Modelle mit gleichem Anforderungsprofil hinsichtlich Telemetriesensoren.
Nun kann man die komplette Klasse mit einem Setup für Ansagen, Resets etc.. konfigurieren, ohne dies jedesmal in die Ethos Konfig modellieren zu müssen.
Mit einer zentralen Änderung im script kann man direkt alle entsprechenden Modelle anpassen.

Man wird unabhängiger vom Sendertyp:
Die Urversion entstammt der Anforderung, **recht schnell** Modelle (unter oTx) von einem Horus X12s auf eine X-Lite pro zu migrieren, **ohne wirklich auf Schaltfunktionen verzichten zu müssen**.
Dazu wurde ein script für die X12 programmiert, das die geforderten „simplen“ Schaltfunktionen über die dort möglichen Eingabedevices abbildete. Ein weiteres script gleichen Namens für die identischen Schaltfunktionen, aber anderer HW Belegung wurde für die X Lite entwickelt.

Kopierte man nun einen Modellspeicher von der X12 auf die X Lite pro hatte man alle diese Funktionen trotz unterschiedlichster HW Konfig sofort parat, ohne irgendeine Anpassung vornehmen zu müssen !
Trick war halt der identische lua script Name !

Hinweise vorab:

(1)

Für das ordnungsgemäße Funktionieren mit Sendergyro setze ich voraus, dass der Sender zuvor unter Ethos korrekt kalibriert wurde.

Ich kalibriere immer so, dass der maximale Gyro Wert erst bei nahezu senkrechter Lageposition erreicht wird.
(Also Sender bei Abfrage der links / rechts Posi etc.. senkrecht halten)

Unter Ethos bis Rev 1.4.4 existiert noch ein Bug in der Kalibrierroutine der X18s/se
Hier bitte issue#2161 beachten !
https://github.com/FrSkyRC/ETHOS-Feedback-Community/issues/2161

(2)

Ethos Lua stellt aktuell (Q4 2022) leider noch nicht alle Spezialfunktionen auch als Lua Methode zur Verfügung.
Die Funktionalität wird aber sukzessive erweitert.

Der Telemetrie reset wird daher aktuell programmatisch nur „nachgebaut“ indem bestimmte Sensoren dediziert resettet werden.

Wirkliche Timer-Resets sind auch nicht möglich, ich setze derzeit den Timer einfach auf 00:00:00.
Der emulierte „reset“ funktioniert daher nur bei aufwärts zählenden Timern.
Auch dazu existieren issues, wird evtl mit 1.4.5 behoben.

Wenn entsprechende Methoden durch Ethos Updates verfügbar werden wird das script entsprechend erweitert werden.

Funktionsweise:

Vier Lageebenenen (standard, links, rechts, vorne) multipliziert mit jeweils sechs möglichen Schaltzuständen per Lage ergeben 24 mögliche Funktionen.
Man muss natürlich nicht alle Funktionen nutzen, je nach Gusto sollte man die Konfig so vornehmen, dass man nicht immer neu nachdenken muss, welche Kombi was auslöst.

Das script liest eine Konfigurationsdatei aus, um zu ermitteln welche Tastdauer in Kombi mit welcher Senderhaltung welche Funktion auslöst.

Nun werden permanent das Sendergyro und die Tastzustände abgefragt.

Der Sender hat in der „Standardhaltung“ einen gewissen Totbereich, damit nicht bei jedem Wackler ein Gyroereignis erkannt wird.
Sobald ein „Schwellwert“ für die Schräglage (ca 30Grad li/re, oder leicht nach vorne) des Senders überschritten wird, wird die Senderlage neu zugewiesen, ein kurzes Tonsignal wird abgesetzt.
Man muss zurück auf neutral um die „Lage“ neu zu definieren, eine direkter Schwenk von „nach vorne“ nach „links“ wird nicht unmittelbar unterstützt, da es sich als schwer handlebar herausgestellt hat.
Optional kann auch für das Wiedererreichen der standard“ Lage ein Tonsignal abgesetzt werden.
(Ist per default aus, störte mich…)

Ausser den typischen Ansage und Reset Funktionalitäten existiert noch die Möglichkeit zwei Logische Schalter jeweils umzuschalten.

Dazu werden immer zwei Bits der sogenannten „Source Variablen“ verwendet.
Die Source Variable kann entsprechend Werte von 0 bis 3 annehmen:

0 >> LSW 1 aus	LSW 2 aus
1 >> LSW 1 ein	LSW 2 aus
2 >> LSW 1 aus	LSW 2 ein
2 >> LSW 1 ein	LSW 2 ein

Um die LSW‘s unter Ethos auzuwerten sind Einstellungen eines LSW‘s nötig, die unter dem Kapitel Konfiguration ausgeführt werden.

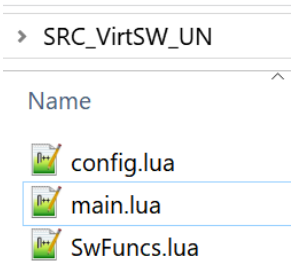
Bedienung

Es handelt sich um ein sogenanntes „source script“ ein wenig analog zum oTx mixer script.
Daher ist keine UI / Bedienung möglich

Installation:

(1) Dateien & Ordner kopieren

Nach download des Zip Files gilt es dieses zu entpacken
Nun steht ein Ordner zur Verfügung, die in das script Verzeichnis des Senders kopiert werden muss:



(2) source script einrichten

Source scripte werden unter Ethos immer in der **Modelleinstellung** aktiviert (s.a. „Edit Model“)

Unter dem entsprechenden Formulareintrag ganzn unten („Lua Sources“) ist das „virtSwitch“ script zu aktivieren:

(3, optional) Logische Schalter einrichten

Soll das script auch dazu verwendet werden logische Schalter einzurichten, müssen diese unter Ethos definiert werden.

Insgesamt werden vier LSW‘s benötigt.

Zwei zur Vorverarbeitung, zwei für die „eigentlicvchen“ Schaltfunktionen.

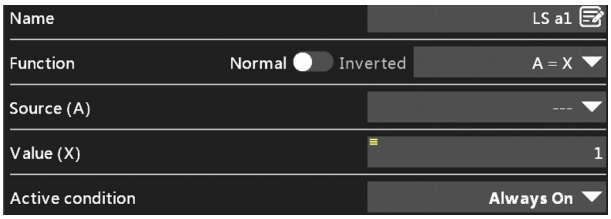
(1) LSW Vorverarbeitung:

Der erste log.Schalter, „**LS A“**, ist aktiv, wenn „der Wert des scriptes“ 1 oder 3 beträgt.

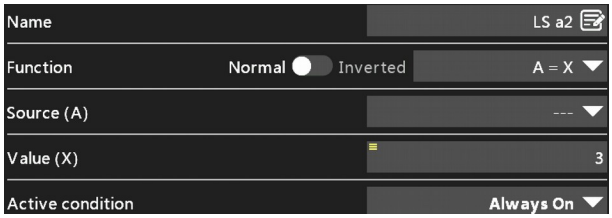
Entsprechend wird ein „LSW a1“ für „Wert =1“ und ein weiterer „LSW a2“ für „Wert =3 definiert

„LS A“ ist aktiv bei der „boolschen Bedingung“: „LS a1“ oder „LS a2“, das macht bereits drei LSW‘s

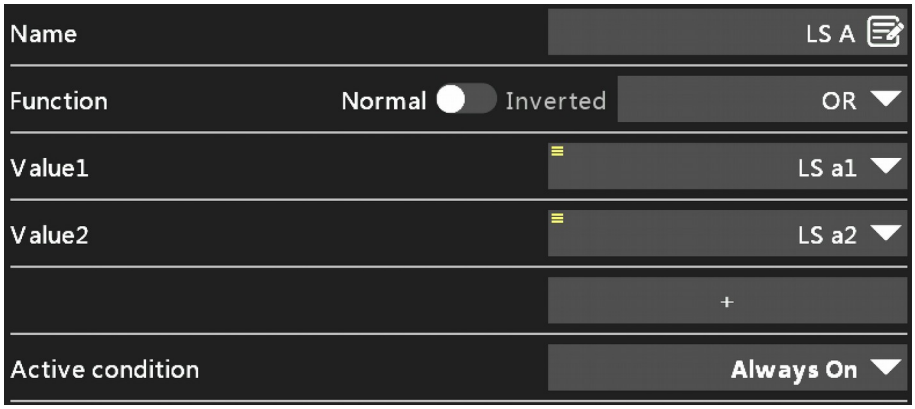
LS a1:



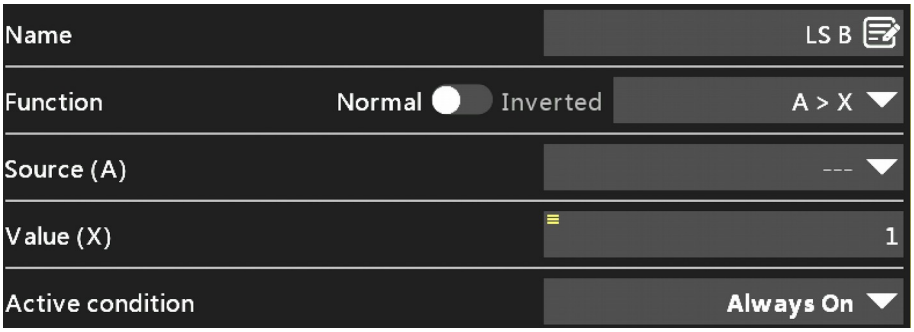
LS a2:



LS A (eigentlicher LSW zur weiteren Verwendung):



LSW2 ist einfach definiert über „Wert > 1“



Konfiguration:

Einleitung:

Die Konfiguration erfolgt über das Editieren von Dateien.
Im Wesentlichen muss die Tasterbelegung in Abhängigkeit der Senderlage definiert werden.

Dies geschieht über Einträge in der Datei „config.lua“
Dort werden die gewünschten Funktionen eingetragen.

Im Umgang mit Rechnern erfahrene Anwender können zudem im Hauptprogramm „main.lua“ Einstellungen für das gewünschte Zeitverhalten der Taster (kurz/mittel/langes Drücken) bzw die Schwellwerte, wie weit ein Sender geschwenkt werden muss, auf die persönlichen Bedürfnisse einstellen.

Konfigtabelle pflegen

Öffnet man die „config.lua“ Datei mit einem Texteditor, erkennt man leicht einen Absatz mit tabellenartigen Einträgen

Eine Zeile entspricht einer Senderlage (normal, links…) eine Spalte einem Tastzustand („links,kurz“ oder „links,mittel“ oder „rechts, lang“ etc..), je nachdem welchen Taster man wie lange gedrückt hält.

Innerhalb der geschweiften Klammern (quasi der „Zelle“) wird nun die gewünschte Funktion, und dann per Komma getrennt ein dazugehöriger Parameter eingetragen.

Eine Telemetrieansage hat die Funktion „playTele“, der gewünschte Sensor in Hochkommata als Text einzutragen. (Hinweis: es muss der englische Begriff der Sensoren eingetragen werden, unabhängig davon auf welche Sprache der Sender konfiguriert wurde)

Liste möglicher Funktionen

Ein Ausdruck in der oben erwähnten Tabelle wie {**"playTele", "Altitude"** } stellt also eine Funktion dar, die per Tastenklick aufgerufen werden kann.
In der Regel besteht ein solcher Ausdruck aus einem Paar:

1. - der eigentlichen Funktion (hier „playTele“), diese muss in Hochkommata gesetzt werden
2. - meistens noch als zweiter Wert einem notwendigen Parameter(Texte müssen auch in Hochkommata gesetzt werden)

Möchte man eine entsprechende Tastfunktion wie „links,lang“ in der Lage „rechts“ z.B. nicht nutzen, sollte eine dummy Funktion eingetragen werden, wie z.B. {**„print“, „right LI“**}, siehe obiges Beispiel.

Wird kein Parameter benötigt, trägt man nil ein, siehe der „resetAlt“ Funktion.

Lua verwendet für die Telemetriesensoren ausschliesslich deren englische Bezeichnung, weiss man diese nicht, empfehle ich im Zweifel folgende Vorgehensweise:

Modell mit möglichst vielen Sensoren als Arbeitskopie kopieren und darauf wechseln
In der Arbeitskopie alle Sensoren löschen
Sender auf englische Bedienung wechseln
Sensoren neu Suchen und Bezeichnungen notieren
Sender wieder auf deutsch anpassen und auf ursprünglichen Modellspeicher wechseln

folgende Funktionen sind aktuell implementiert:

playTele, sensor	Ansage des Sensorwertes „sensor“ (sensor = text)
playTmr, timer	Ansage des Timers (timer = Nummer)
resetTele, nil	reset Telemetrie (aktuell workaround)
resetAlt, nil	reset Höhe
resetTmr, timer	reset timer (timer = Nummer; setzen auf Wert 0)
toggle, num	Umschalten eines LSW‘s (num = Numer 1 oder 2; interne LSW Nummer)

Hinweis für Lua Enthusiasten:

Diese „Paare“ werden quasi als Funktionsaufrufe direkt über den globalen Namensraum „durchgereicht“, das ermöglicht auch Aufrufe direkter Lua Methoden/Funktionen, falls diese gewünscht werden (siehe „print“) ...

Sounds

Ich mag es gerne „kurz und knackig“, standardmäßig wird bei der Ausgabe von Telemetriewerten daher nur der Wert ohne weitere Sprachansagen gemeldet.
Wer einen „Präfix“ wie „Höhe“ , „Entfernung“, „Empfängerspannung“ usw mit in der Ansage enthalten haben möchte kann dies reinkonfigurieren.

Dies erfolgt ebenfalls über die Datei „config.lua“

zunächst wird der „Pauschale Konstante“ namens soundfile dazu auf true gesetzt:

```
--    sounds
local soundfiles <const>        = true
```

In einer weiteren Tabelle werden den Telemetriesensoren (labels) die dazugehörige Sound Datei zugewiesen:

```
--                              label/sensor        folder              file
local calls                     = {
                                Altitude        =   audiofld    ..  "alti.wav" ,        -- "Alti"
                                Dist            =   audiofld    ..  "dist.wav" ,        -- "Dist"
                                timer1          =   audiofld    ..  "T1.wav" ,          -- "Timer"
                                timer2          =   audiofld    ..  "T2.wav" ,
                                timer3          =   audiofld    ..  "T3.wav" ,
                                resetT1         =   audiofld    ..  "reset_t1.wav" ,    -- "reset timer"
                                resetT2         =   audiofld    ..  "reset_t2.wav" ,
                                resetT3         =   audiofld    ..  "reset_t3.wav" ,
                                resetAlt        =   audiofld    ..  "reset_alt.wav",
                                resetTel        =   audiofld    ..  "reset_tel.wav",
                                }
```

Bei Aufruf eine Telemetriesensors, oder Ausführung einer Funktion (reset…) wird diese Sounddatei dann als erstes aufgerufen. Man muss nur die Dateinamen zu den Labels definieren, bei denen eine Vorabansage gewünscht wird.
Ist keine Sounddatei definiert, wird nur der Wert angesagt.

Sounds muss man sich den individuellen Ansprüchen gemäß selber generieren.
Ich empfehle das Programm TTSAUTOMATE.
Zugehörige psv Dateien als Beispiel zum Erstellen eigener Soundfiles sind im Ordner „ttsautomate“ zu finden.

Die Sounddateien an sich gehören in den entsprechenden ETHOS-Audioordner, direkt unter dem Ordner für die jeweilige Sprachversion (also z.B. /audio/de)

Ethos verwendet für
die deutsche Sprache die Einstellung „de-DE-Wavenet-E"
die englische Sprache die Einstellung „en-GB-standard A“

Aller Voraussicht nach wird das neue TTSAutomate (in Entwicklung, Stand Q4 2022) diese Sprachen unterstützen.

Sonstige Optionen

Wer es sich zutraut kann in der main.lua weiteres Feintuning vornehmen.

Neigungs Schwellwerte

Möglich sind zum Beispiel das Setzen des Schwellwertes, ab welcher Schräglage des Senders ein dazugehöriges „Gyro Event“ getriggert werden soll.

Üblicherweise liefert der Sender Werte zwischen -1024 (z.B. Senkrecht in die eine Extremposition) bis +1024 (Senkrecht in die andere Extremposition), wobei die Extremposition abhängig der vorgelagerten Kalibrierung ist.
Neutralstellung liegt ca bei 0, kleine Offsets sind möglich.

Für den Schwenk links/rechts definiere ich einen „Totbereich“, der die Neutral-Lage definert.
Wird der Grenzwert überschritten, geht das Script in den jeweils gewählten „Gyro Modus“.

Der Totbereich wird in der Variablen“X_deadzone“ definiert

```
25  -- attitude
26  local X_deadzone <const>            = 600
27  local Y_fwd_active <const>          = 300
```

Analog dazu definiert „Y-fwd_active“ den Schwellwert für das nach vorne Neigen.

Tastzeiten

Die für die Unterscheidung „kurzes/mittleres/langes Drücken“ notwendigen Parameter sind folgende:

```
21  -- timing
22  local press_short <const>           = 0.3
23  local press_long <const>            = 0.8
24
```

Tastzeiten < 0.3 Sekunden werden dadurch als „kurz“, Zeiten >0.8 Sekunden als „lang“ definiert, dazwischen ergo „mittel“. Diese Parameter können individuell angepasst werden.

Akustik-feedback

```
15  local BEEP_leaveNeutral <const>     = true
16  local BEEP_back2Neutral <const>     = false
```

Diese beiden Parameter können jeweils auf „true“ oder „false“ gesetzt werden.
Der erste Parameter ermöglicht einen Signalton bei Überscheiten des Gyro-Schwellwertes, der zweite bei Wieder-Erreichen der Neutralstellung

Rev 0.8
unow, Dez 2022