

Schaltpult für OpenTX und EdgeTX Sender

V 0.3.a



Generelles

Dies ist mein privates Bastelprojekt.

Jeder ist eingeladen das Projekt nachzubauen.

Ich übernehme keine Gewähr für in diesem Zusammenhang getätigte Angaben. Eine Haftung für Schäden, die durch den Betrieb entstehen ist ausgeschlossen.

Die aktuellen Dokumente und Software findet ihr hier:

<https://github.com/Tiefflieger68/Schaltpult-Arduino-Nano>



Beschreibung

Entwickelt wurde die Schaltung für die Möglichkeiten von OpenTX und EdgeTX Fernsteuerungen.

Anschlüsse für bis zu 14 zusätzliche Schalter und 2 Potis.

Schalteingänge mit diesen Schalterkonfigurationen sind, beliebig kombiniert, nutzbar:

- 2-Pos-Schalter
- 3-Pos-Schalter
- Einfache Taster
- Doppeltaster (2 Taster auf einem Eingang)

Die Schaltung erzeugt ein 16-Kanal PPM oder SBUS Signal.

PPM wird über die Trainer-Buchse oder den externen Modulschacht,

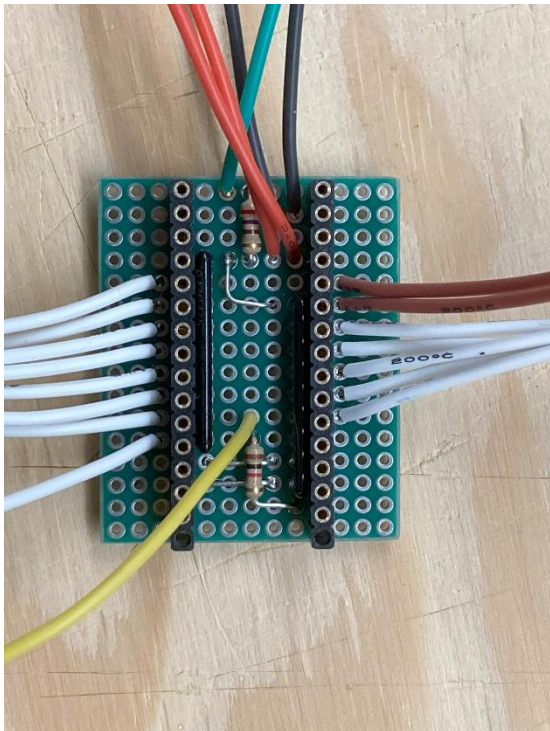
SBUS wird über den Serial Port oder den externen Modulschacht in den Sender eingespeist.

Schaltung

Die Schaltung wurde bewusst sehr einfach gehalten und kann mit wenig Aufwand auf einer Lochrasterplatine aufgebaut werden. Die meisten Verbindungen werden über nebeneinanderliegende Lötpins hergestellt.

Für die 100kOhm Pulldown-Widerstände benutzt man sinnvollerweise Widerstandsnetzwerke ("SIL 9-8 100K") um den Verdrahtungsaufwand zu minimieren.

siehe „Schaltplan.png“



„Mainboard“ mit allen Anschlusskabeln und das fertige Pult für die RM MT12

Anschaltung des Schaltpults an den Sender

Achtung: unbedingt darauf achten, dass entsprechend der Spannungsquelle, der richtige Spannungseingang des Arduino-Boards genutzt wird.

Je nach Protokoll, stehen verschiedene Anschlussmöglichkeiten zur Verfügung und es ist der entsprechende Anschluss am Arduino Nano auszuwählen.

- PPM: 16-Kanal PPM Signal
- SBUS: Standard SBUS (Modulschacht)
- SBUS UART: SBUS für non-inverted UART (Serial Ports)

siehe „Anschaltung 0.3.x.pdf“

Nicht alle Möglichkeiten stehen bei allen Sendern zur Verfügung.

Wenn möglich sollte eine Variante mit SBUS-Protokoll gewählt werden, da die Werte digital und damit genauer übertragen werden.

Code und Binaries

Der Code wurde mit Bascom-AVR 2.0.8.6 erzeugt.

Zusätzlich wird ein Bugfix für „serout“ benötigt.

https://www.mcselec.com/index2.php?option=com_forum&Itemid=59&page=viewtopic&p=84137#84137

Siehe "Source" und "Binary"

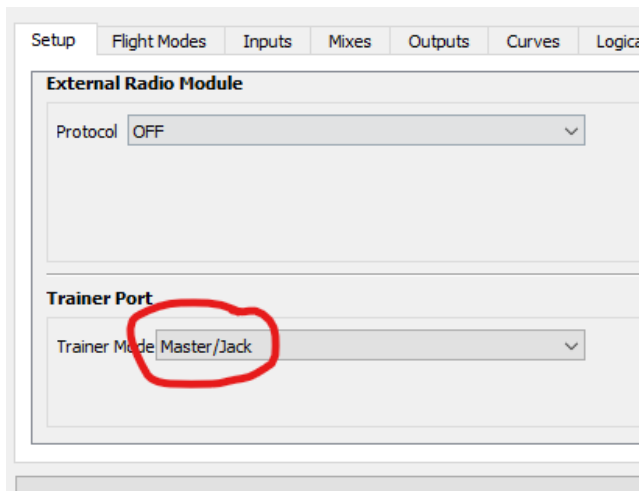
Einstellungen für OpenTX und EdgeTX

Das Schaltpult wird wie ein Schüler-Sender, bei einer Schüler-Lehrer-Verbindung angeschlossen.

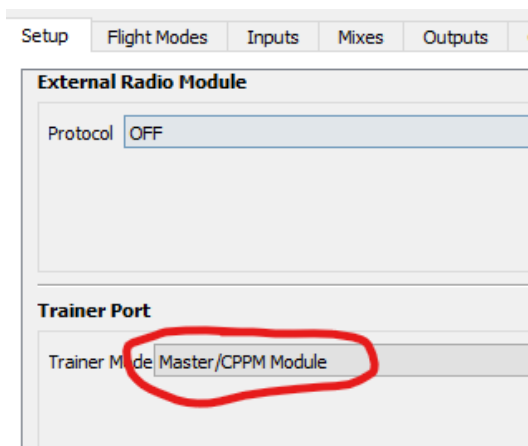
Das Signal (TR1 – TR16) kann direkt in den Mischern und logischen Schaltern verarbeitet werden.

Allgemeine Modell Einstellungen:

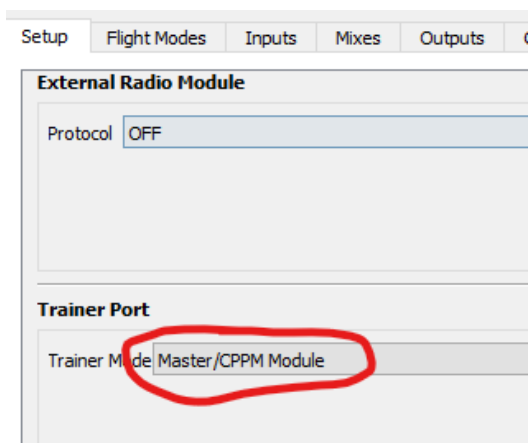
Nicht alle Möglichkeiten stehen bei allen Sendern zur Verfügung. Siehe „Anschaltung an den Sender“
PPM-Signal über Trainer-Buchse:



PPM-Signal über Modulschacht:

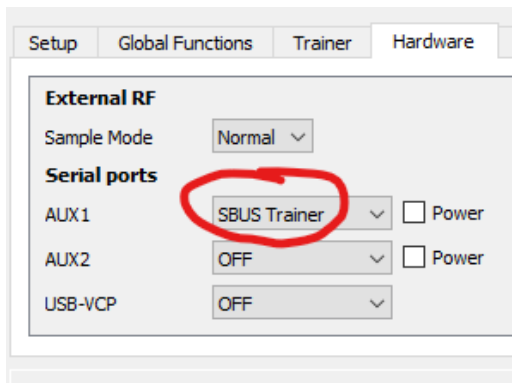


SBUS über Modulschacht:

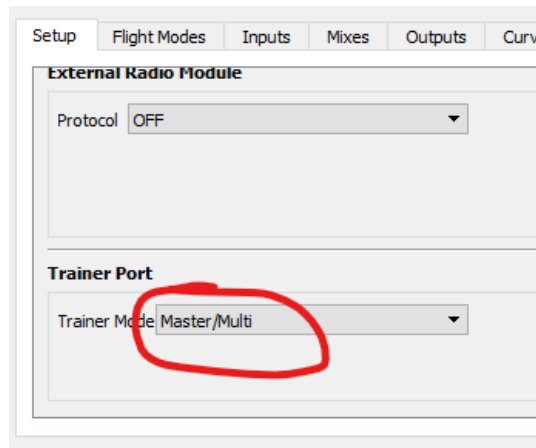


SBUS über Serial Port:

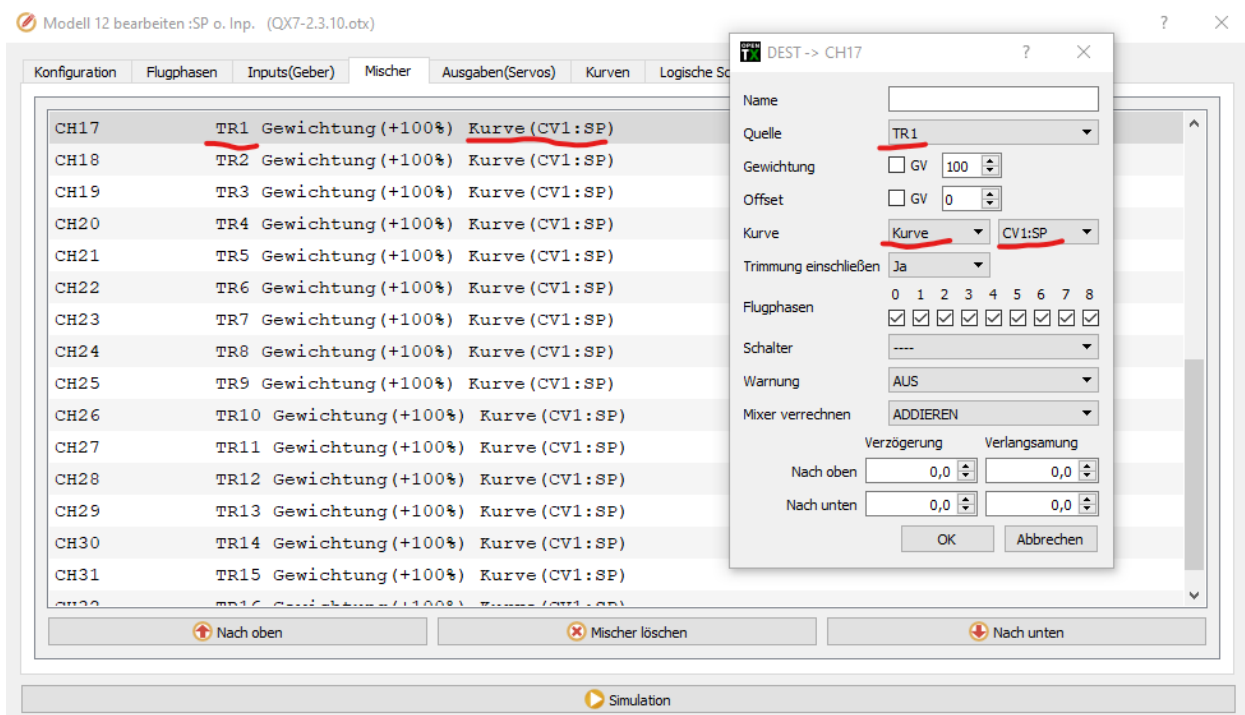
System Settings:



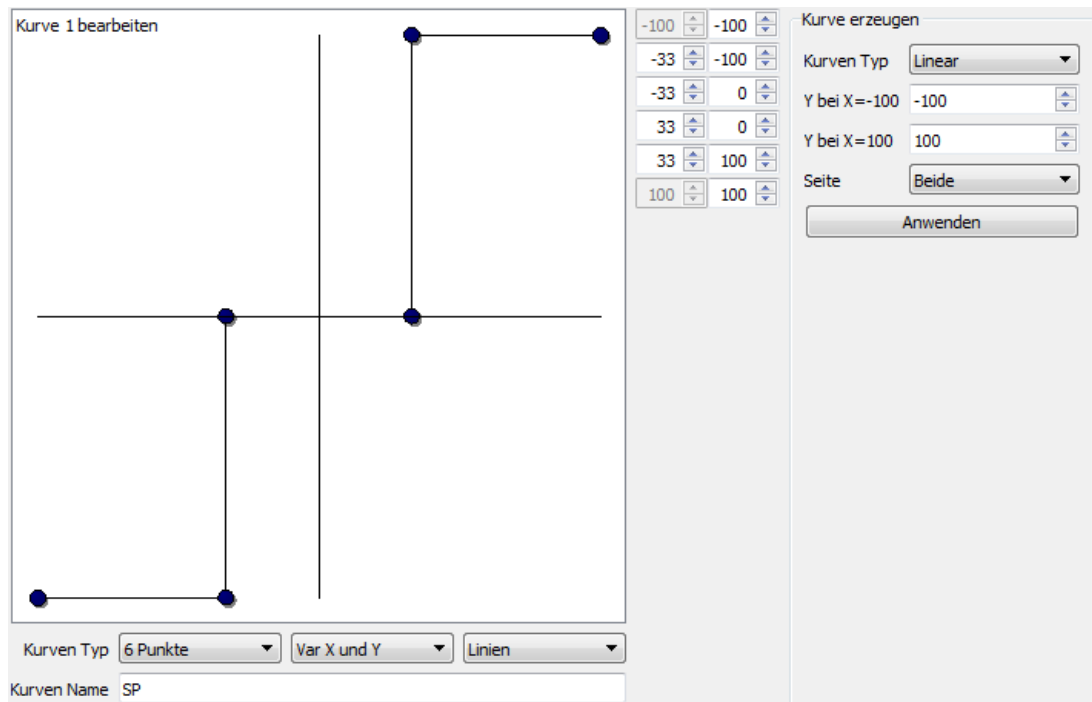
Model Settings:



Beispiel für Auswertung in Mischern:



Bei Verwendung des PPM-Signals entstehen Ungenauigkeiten beim Einlesen des Signals und es werden die Werte -100%, 0%, +100% nicht immer exakt getroffen. Man kann das Signal einfach mit einer „Kurve“ aufbereiten. Nutzt man das SBUS Signal, ist keine Kurve nötig.



Hier wird Schalter 1 auf Kanal 17 mit exakten Werten -100% , 0% , +100% ausgegeben

Beispiel für Auswertung in logischen Schaltern:

Konfiguration					
Flugphasen					
Inputs(Geber)					
Mischer					
Ausgaben(Servos)					
Kurven					
Logische Schalter					
Spezial Funk					
#	Funktion	V1	V2		
L01	$a > x$	TR1	30		----
L02	$a < x$	TR1	-30		----

Im Beispiel wird der erste 3-Pos-Schalter von dem Pult ausgewertet.

Schalter nach oben: L01 aktiv

nach unten: L02 aktiv

Mittelstellung: weder L01 noch L02 aktiv

Kalibrieren der Poties:

Durch Fertigungstoleranzen wird in der Mittelstellung des Potis evtl nicht der Wert „0%“ ausgegeben. Mechanisch bedingt kann beim Endanschlag des Potis evtl nicht der Wert „-100%“ oder „100%“ erreicht werden.

Die kann mit einer Kurve ausgeglichen werden. Es ist sinnvoll diese Einstellung direkt am Sender vorzunehmen, da man dann gleich das Ergebnis sieht.

INPUTS	Pot
Input	Pot 0.0
Name	---
Source	TR16
Weight	100
Offset	0
Curve	
Cstm	CV1

Ein Input erstellen.
Quelle: das Poti (TR15/16)
Auf Kurve verweisen (hier CV1)

Poti betätigen, Endausschläge und Mitte ermitteln (hier -58.8, 62.5, 10.5)

INPUTS	Pot
Input	Pot 0.0
Name	---
Source	TR16
Weight	100
Offset	0
Curve	
Cstm	CV2

-58.8

INPUTS	Pot
Input	Pot 0.0
Name	---
Source	TR16
Weight	100
Offset	0
Curve	
Cstm	CV2

62.5

INPUTS	Pot
Input	Pot 0.0
Name	---
Source	TR16
Weight	100
Offset	0
Curve	
Cstm	CV2

-10.5

CURVES	CV1
Name	---
Type	Custom
Count	5
5pts	
Smooth	<input type="checkbox"/>

5-Punkt-Kurve erstellen
Typ: Custom

Ermittelte Werte des Potis für Endausschläge und Mitte in Pt 2, 3, 4 übernehmen

CURVES	CV1
Name	---
Pt2	x=-58 y=-100
5pts	
Smooth	<input type="checkbox"/>

CURVES	CV1
Name	---
Pt3	x=-10 y=0
5pts	
Smooth	<input type="checkbox"/>

CURVES	CV1
Name	---
Pt4	x=60 y=100
5pts	
Smooth	<input type="checkbox"/>

Das war's schon

Spannungsabgriff am externen S.Port

Um den externen S.Port für die Spannungsversorgung nutzen zu können, muss „S.Port Power“ in den Hardwareeinstellungen aktiviert werden. (In Companion gibt es den Punkt erst ab openTX 2.3.11.)

