

# Schaltpult für OpenTX Sender

Version 1.0

## **Beschreibung:**

Eingänge für 16 zusätzliche Schalter.

16 Schalteingänge mit diesen Schalterkonfigurationen sind, beliebig kombiniert, nutzbar:

- 2-Pos-Schalter oder Taster mit Wechselkontakt. Ausgabewert: -100%, +100%
- 3-Pos-Schalter oder Taster mit Wechselkontakt. Ausgabewert: -100%, 0%, +100%
- Doppeltaster (2 Taster auf einem Eingang/Kanal). Ausgabewert: -100%, 0%, +100%
- Schalter oder Taster mit Schließerkontakt mit Pullup-Widerstand. Ausgabewert: -100%, +100%
- Schalter oder Taster mit Schließerkontakt ohne Pullup-Widerstand. Ausgabewert: -100%, 0%

Unbenutzte Eingänge bleiben unbeschaltet. Siehe auch Beschaltungsbeispiele im Schaltplan.

Die Schaltung erzeugt ein 16-Kanal PPM Signal, abhängig von der jeweiligen Schalterstellung.

Dies wird über die Trainer-Buchse oder den Modulschacht in die Fernsteuerung eingespeist.

Die Versorgungsspannung muss 7 – 14 V betragen.

Entwickelt wurde die Schaltung für die Möglichkeiten von OpenTX Fernsteuerungen.

Evtl. können aber auch andere Systeme das PPM-Signal nutzen.

Aktuell getestet mit:

FrSky Taranis Q-X7 OpenTX 2.2.1

Trainer-Buchse: OK

Modulschacht: OK

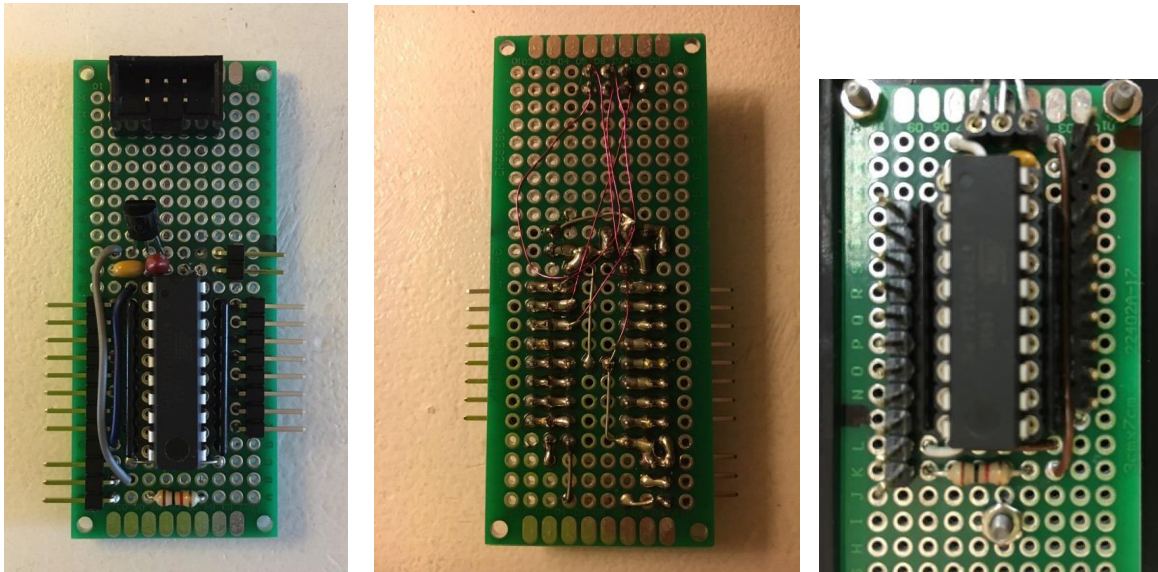
FrSky Taranis X9D+ OpenTX 2.1.19

Trainer-Buchse:

Modulschacht:

## **Schaltung:**

Die Schaltung wurde bewusst sehr einfach gehalten und kann mit wenig Aufwand auf einer Lochrasterplatine aufgebaut werden. Auf der Platine sind nur wenige Kabelverbindungen nötig. Die meisten Verbindungen werden über nebeneinanderliegende Lötpins hergestellt.



v.l.n.r.: Prototyp mit ISP, dto Rückseite, finales Board

Der ISP –Stecker (6-Pol Stecker oben) ist nur optional und für den normalen Betrieb nicht nötig. Entsprechend entfallen dann auch die roten Fädel-Drähte. Ohne ISP Stecker ist die Platine ca. 4 x 2 cm groß.

Ein Spannungsregler im TO-92-Gehäuse ohne Kühlung ist ausreichend.

## **Daten:**

Controller:	ATTiny2313 @4MHz
Eingangsspannung:	7 – 14 V
Stromverbrauch:	5 - 6 mA

## **Funktion:**

Ein AVR Microcontroller (ATTiny2313) liest über 16 Eingänge die 3 möglichen Schaltzustände (+, -, offen) je Schalter ein. Um 3 verschiedene Schaltzustände erkennen zu können sind die beiden 100kOhm Widerstandsnetzwerke nötig (links und rechts vom AVR).

Abhängig von den erkannten Schaltzuständen wird dann das PPM-Signal generiert.

Der Widerstand (unten) schützt die Schaltung bei einem Kurzschluss am PPM-Port z.B. beim Einstecken des Steckers in die Trainer-Buchse.

## Einstellungen für OpenTX (Companion 2.2.1):

Das externe PPM-Signal (TR1 – TR16) kann nur in den Inputs ausgewertet und nicht direkt in Mischern oder Logischen Schaltern verarbeitet werden. Daher muss für jeden Kanal ein Input erstellt werden.

Das Beispiel gibt die Schalter auf Kanal 17-32 aus.

### Allgemeine Modell Einstellungen:

Name: Schaltpult

Trainer Port: Lehrer/CPPM Module

### Inputs:

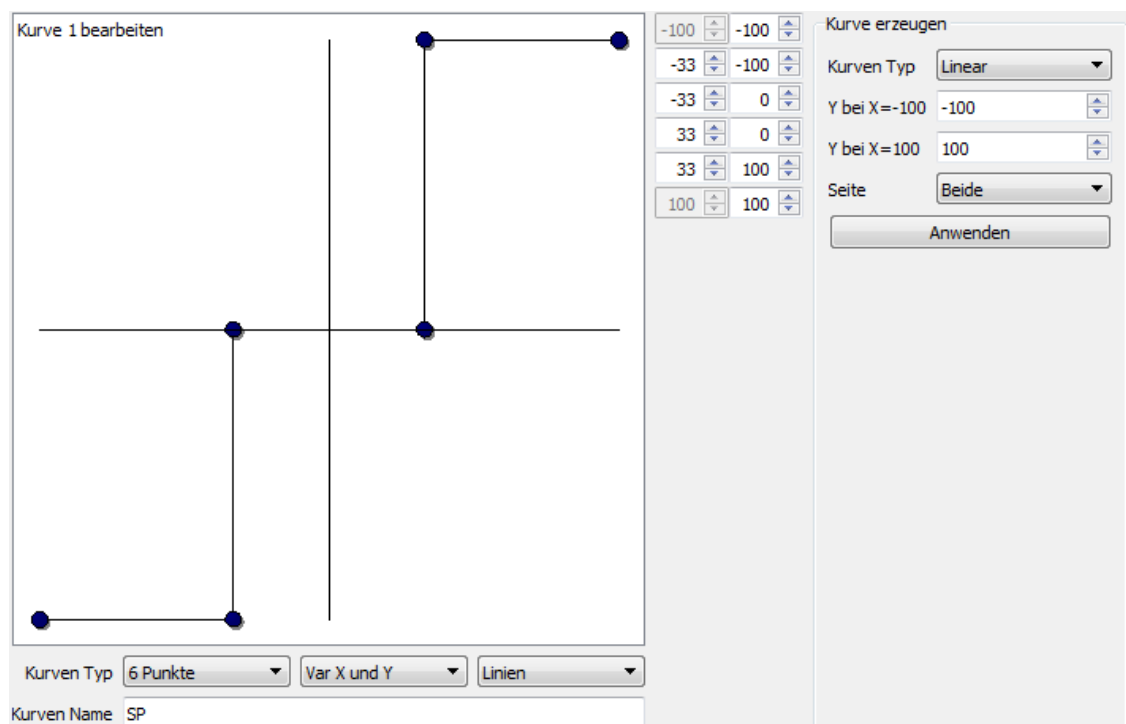
I17	TR1	Gewichtung(+100%)	Kurve(CV1:SP)
I18	TR2	Gewichtung(+100%)	Kurve(CV1:SP)
I19	TR3	Gewichtung(+100%)	Kurve(CV1:SP)
•			
•			
•			
I32	TR16	Gewichtung(+100%)	Kurve(CV1:SP)

### Mischer:

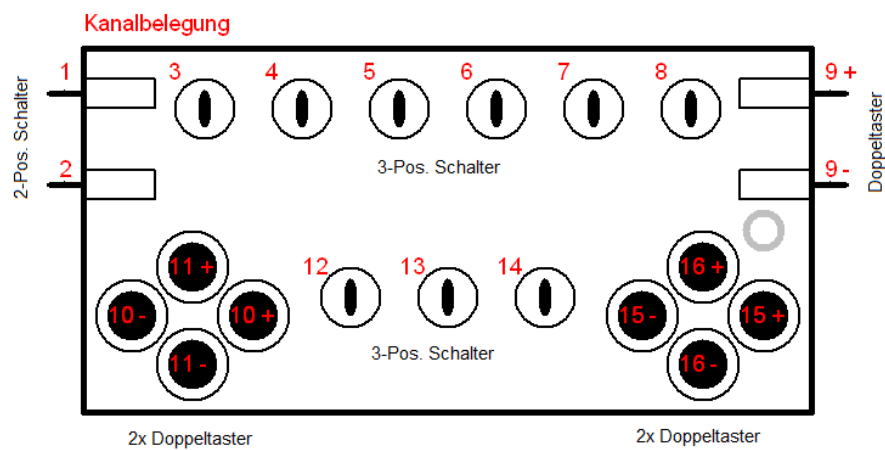
CH17	I17	Gewichtung(+100%)
CH18	I18	Gewichtung(+100%)
CH19	I19	Gewichtung(+100%)
•		
•		
•		
CH32	I32	Gewichtung(+100%)

Da die Schaltung ohne externen Quarz aufgebaut wird und aufgrund der Serienstreuung der AVR-Microcontroller, werden die Werte -100%, 0%, +100% nicht immer exakt getroffen. Ggf. sind Anpassungen in den Inputs oder Mischern nötig.

Dies kann am einfachsten mit der Integration einer „Kurve“ in den Inputs geschehen



## Mein erstes Schaltpult



Meine erste Umsetzung. Ein Vollausbau mit 16 genutzten Kanälen, wie hier, ist für viele Projekte wohl oversized. Zumal ein nicht unerhebliches Gewicht zusammenkommt.