

# MultiSwitch – Funktionsübersicht und Änderungen der Versionen (Stand 18.11.2021)

## Funktionsübersicht

- Variante 1: Mit 1 Taster / Prop. Kanal können max. 11 dig. Kanäle gesteuert werden
- Variante 2: Über einen Jumper können 4 Kanäle direkt als Leistungsausgang geschaltet werden. Die Schaltkanäle von Variante 1 werden dann auf 7 Kanäle verkleinert.
- Variante 2 kann über einen Jumper aktiviert/deaktiviert werden.
- Pro Arduino können max. 1A direkt auf dem Print über die Transistoren geschaltet werden.
- Max. 6A für den gesamten Print
- Die Vorwiderstände R12 bis R22 sind noch zu berechnen bei der Verwendung von LED's.
- Spannungsmessung für Ext. Spannung mit Aktivierung des Schaltausganges

## Funktionsbeschreibung

Variante 1:

Bei Variante 1 ist der Jumper JP17 zwischen Pin 1 und 2 gesteckt und der Eingang auf 0V geschaltet. Über einen Taster (min. oder max. PWM Ausschlag) wird ein Counter hochgezählt und beim entsprechenden Ausgang das anliegende Signal invertiert geschaltet. Der Schaltausgang wird nach 1.5 Sekunden Inaktivität beim Countereingang geschaltet. Der Counter wird nach dem Schaltvorgang wieder auf 0 zurückgesetzt.

Variante 2:

Variante 2 funktioniert grundsätzlich gleich wie Variante 1 jedoch werden nur 7 Ausgängen mit der Tastfunktion bedient. Hinzu kommen vier Ausgänge (Klemme X1) welche direkt über vier Eingänge geschaltet werden. Die dazugehörigen Eingänge können über Servokabel direkt ab dem Empfänger über die Jumper JP13-JP16 verkabelt werden. Die Aktivschaltung erfolgt über den Jumper JP17 mit einer Brücke zwischen Pin 2 und 3 (+5V).

Bei beiden Varianten kann bei jedem Ausgang direkt ein spezielles Arduino Leistungsrelais angesteuert werden, falls grössere Leistungen (grösser 1A) geschaltet werden müssen. Oder es wird eine Transistorschaltung verbaut welche direkt bis max. 1A pro Kanal (max. 6A für den Print) angesteuert werden können (z.B. LED-Beleuchtung). Die Ansteuerung des Counters und den direkten Eingängen können über prop. Kanäle oder eines Steuerknüppels erfolgen.

## Spannungsmessung inkl. Abschaltung

Die Funktion der Spannungsmessung inkl. Schaltausgang erklärt sich am besten über ein Beispiel (weiter unten). Die Funktion wird über den Jumper J18 aktiviert bzw. deaktiviert. Wird die Funktion **nicht** benötigt, muss **zwingend** der Jumper auf **GND** gesteckt sein. Beim Unterschreiten einer Sollspannung wird ein Ausgang geschaltet und es steht für Variante 1 und 2 jeweils ein Ausgang weniger zur Verfügung.

Variante 1: Arduino Ausgang D13 bei Klemme X1-1

Variante 2: Arduino Ausgang D9 bei Klemme X2-1

Die Spannung wird standardmässig ca. alle 10 Sekunden gemessen. Wird der eingestellte Spannungswert nacheinander mehr als den definierten Wert unterschritten, wird der Ausgang aktiviert. Wird der Wert nur einmalig unterschritten und bei der nächsten Messung ist Wert wieder grösser, wird der Counter wieder zurückgesetzt.

Die beiden Widerstände im Schema R26 und R27 müssen ausgemessen werden. Die Werte müssen möglichst genau in den Arduino Code übernommen werden.

Folgende Variablen können im Arduino Code angepasst werden:

setSpannungMin	Wert für minimale Spannung.
countAnzMaxSpanMin	Counter für die Anzahl der Unterschreitungen bis der Ausgang schaltet.
TimeSpannungMessung	Zeit in Sekunden zwischen den Spannungsmessungen, Standard ist 10 Sekunden.
R26	Widerstandswert
R27	Widerstandswert

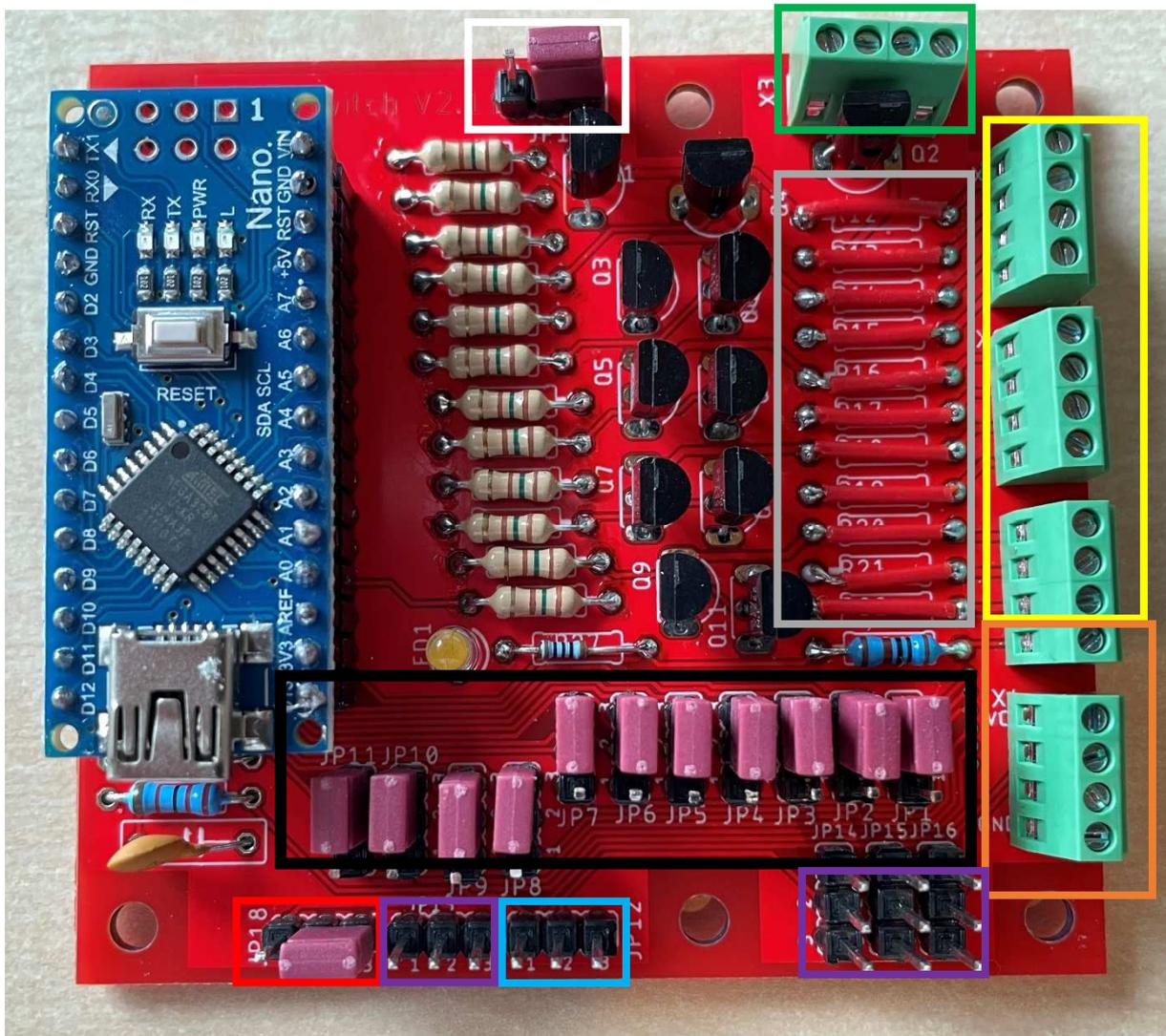
Beispiel:

Neben dem Fahrakku wird ein eigenständiger Akku für die Sonderfunktionen verwendet. Der Fahrakku kann über die Telemetrie ausgelesen werden, der zweite Akku jedoch nicht. Die Spannungsmessung soll eine zu tiefe Akkuspannung verhindern und schaltet beim Unterschreiten eines einstellbaren Limits (im Arduino Code) einen Ausgang auf HIGH. Der Ausgang wird über den Jumper direkt verbunden und schaltet ein Relais welches den Stromkreis des Zusatz Akkus unterbricht.

**Wichtiger Hinweis:**

Die Empfängerspannung muss das gleiche Potential (gleiche Spannung) wie diejenige vom Arduino haben. Ist dies nicht der Fall funktioniert der Print nicht!

## Beschrieb Print Klemmen-/Stiftleisten Anschlüsse



Der Beschrieb richtet sich nach den farblichen Markierungen:

Grün	Je zwei GND (links) und +5V vom Arduino (rechts) Anschlüsse.
Grau	In diesem Fall die überbrückten Widerstände. Hier könnten Widerstände für LED-Beleuchtungen bestückt werden.
Gelb	Klemmen für die max. 11 Schaltausgänge. Beim Bild oben ist die unterste Klemme der erste Ausgang, wenn beim Counter hochgezählt wird. Ist Jumper J17 auf VCC gesetzt, sind die obersten vier Ausgänge direkt geschaltet.
Orange	Klemmen von oben nach unten: <ul style="list-style-type: none"> <li>- Externe Spannungsmessung (falls aktiv)</li> <li>- +5V vom Arduino</li> <li>- Klemme für einzeln Anschluss der Counterfunktion (parallel zu J12)</li> <li>- U<sub>BEC</sub> =&gt; Spannungsversorgung vom Empfänger, meistens ein ext. BEC</li> <li>- GND der Spannungsversorgung vom Empfänger</li> </ul>
Rot	Jumper für Aktivierung/Deaktivierung der ext. Spannungsmessung.
Blau	J12 Steckplatz für Servokabel ab Empfänger für die Counterfunktion.
Schwarz	Jumper für die Überbrückung der NPN Transistoren für direkte Ansteuerung eines Relais bei den Ausgängen 1-11.
Violett	Steckplätze für Servokabel für die direkte Schaltung der Ausgänge 8-11 falls J17 auf VCC gesteckt.
Weiss	J17 für Aktivierung der direkte Ausgänge 8-11

## **Änderungen Hardware**

V2.13

- Domain auf PCB Layout hinzugefügt

V2.11 V2.12

- Entwicklungsvarianten

V2.10

- Erweitert mit Spannungsmessung

V2.8 und V2.9:

- Entwicklungsvarianten

V2.7

- Taster Signal für das Hochzählen mit einem Empfängerkabel über auf eine Stiftleiste gesteckt werden.
- Kleinere Verbesserungen (Bauteil Umplatzierung) am Layout

V2.5

- HW Korrektur nach Fehler beim Design

V2.4

- Entfernung der vorhandenen Feinsicherung
- Erweitert vier Analogen Eingängen und zwei Jumper (JP12 und JP17) für Erweiterung mit direktem Durchschalten von Eingängen.

V2.2 und V2.3

- Entwicklungsstufen ohne Printproduktion

V2.1

- Erweitert auf 11 digitale Ausgänge
- Korrektur Arduino Betriebsspannung auf direkter Verdrahtung. Spannung gemäss Datenblatt möglich

V1.0

- Erste Version mit Schaltung von 8 digitalen Ausgängen über eine Taster Funktion
- Arduino Speisung bei 5V

## **Änderungen Software**

V2.4

- Korrektur beim Einlesen des Auswahl Jumpers von A6 auf A1
- Codeverbesserungen

V2.3

- Erweitert für das Durchschalten der vier analogen Eingänge auf die Ausgänge
- Auslesen der Jumper für die Aktivierung

V2.2

- Codeverbesserungen

V2.1

- Erweitert für 11 digitale Ausgänge
- Startsequenz mit LED-Flackern integriert

V2.0

- Entwickler Stufe

V1.0

- Erste Variante für den Counter mit 8 digitalen Ausgängen.