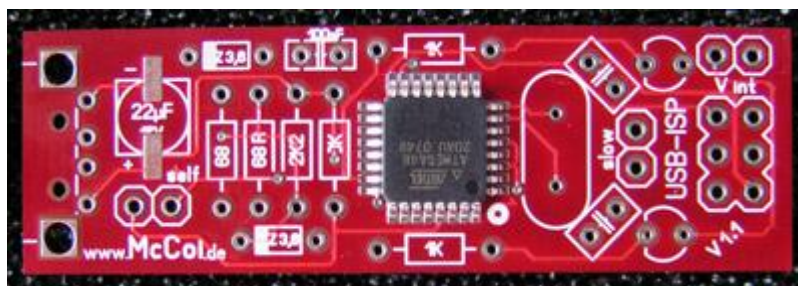


ISP-Stick

Stand Juli 2015
Copyright © by Steffen Koberitz

Der ISP-Stick ist eine USB-Schnittstelle zum Programmieren des McCoi.

Nachdem die parallele Schnittstelle an modernen Rechnern immer seltener wird und auch die serielle Schnittstelle nicht unbedingt als Standard zu betrachten ist, wurde es Zeit, daß nun auch der McCoi über USB programmiert werden kann. Da kommerzielle Lösungen wie z.B. der *Atmel AVR-ISP MK2* weder vom Konzept noch vom Preis zum McCoi passen, habe ich mich umgesehen und eine sehr schöne Lösung gefunden, die sowohl preislich als auch vom Konzept her gut mit dem McCoi harmoniert.



Der vorliegende Bausatz mit dem Namen **ISP-Stick** enthält zwar zwei Bauteile im SMD-Format, doch das soll den geeigneten McCoi-User nicht abschrecken. Der Controller mit seinen 32 Beinchen im TQFP-Format ist bereits vorprogrammiert und auf die Platine gelötet! Damit bleiben nur noch die üblichen, großen Bauteile übrig, wie sie auch beim McCoi verlötet werden. Wer also den McCoi erfolgreich aufgebaut hat (oder aufbauen wird), sollte mit dem ISP-Stick keinerlei Probleme haben.













Damit alles in das schöne USB-Gehäuse paßt, wurde auch der Elko im SMD-Format gewählt. Diesen aufzulöten ist aber aufgrund seiner Größe recht einfach, was später noch erklärt wird.

Kommen wir zunächst zu den Bestandteilen des Bausatzes.


Stückliste

Anz.	Wert	Bauteil	Bezeichn.	Kommentar
2	68 Ω	Widerstand	R1, R2	(schwarz) - blau - grau - schwarz – braun
2	1 kΩ	Widerstand	R3, R4	braun – schwarz – schwarz – braun – braun
1	2,2 kΩ	Widerstand	R5	rot – rot – schwarz – braun – braun
1	10 kΩ	Widerstand	R6	braun – schwarz – schwarz – rot – braun
2	33pF	Kondensator	C1, C2	Aufdruck beachten! braun, gelb oder mehrfarbig - "33CH"
1	100nF	Kondensator	C3	meist blau oder orange, immer glänzend - "104"
1	22µF/16V	Elko	C3	SMD! Minus ist schwarz markiert
2	ZD-3,6	Zener-Diode	D1, D2	Polung beachten
2	20mA	LED, klar	D3, D4	Leuchtdioden klares Gehäuse, leuchtet rot), Polung beachten langes Beinchen ist Plus und muß zur Stiftleiste zeigen!
1	12 MHz	Quarz	Q1	
1	TQFP 32	IC ATmega48	IC1	Controller vorprogrammiert und verlötet
1	USB-A	USB-Stecker	USB-A	zum Anschluß an einen Computer
1	2x3	Stiftleiste 6-polig	Con06	zum Anschluß an den ISP-Stecker des McCoi

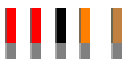
Widerstand-Farbcodes

Farbe	Wert für 1. Ring	Wert für 2. Ring	Wert für 3. Ring	Wert für 4. Ring	Wert für Toleranzring
schwarz 	0	0	0	x1	--
braun 	1	1	1	x10	+/- 1%
rot 	2	2	2	x100	+/- 2%
orange 	3	3	3	x1000	--
gelb 	4	4	4	x10.000	--
grün 	5	5	5	x100.000	--
blau 	6	6	6	x1.000.000	--
violett 	7	7	7	x10.000.000	--
grau 	8	8	8	x100.000.000	--
weiss 	9	9	9	x1.000.000.000	--
gold 	--	--	--	--	+/- 5%
silber 	--	--	--	--	+/- 10%
ohne	--	--	--	--	+/- 20%

Beispiel 1

 bedeutet 2-2-0 x1 1% also
 220 Ohm mit 1% Toleranz

Beispiel 2

 bedeutet 2-2-0 x1000 1% also
 220.000 Ohm (= 220kOhm) mit 1% Toleranz

Beispiel 1

bedeutet 2-2-0 x1 1% also
220 Ohm mit 1% Toleranz

Beispiel 2

bedeutet 2-2-0 x1000 1% also
220.000 Ohm (= 220kOhm) mit 1% Toleranz

Zur Frage, wie unterscheide ich den ersten Ring vom Toleranzring, also die Frage, wie herum man so ein Teil hält: Der Toleranzring ist immer ein Stück weit abgesetzt von den anderen Ringen! Man sieht es auch in den beiden Beispielen. Wobei im Beispiel 1 der Abstand bewusst größer gewählt wurde. Aber auch im Beispiel 2 ist der Abstand des braunen Rings größer als der zwischen den anderen! Die dicke des Rings hat nichts zu sagen...!

Aufbau der Platine

Nachdem das schwierigste Bauteil -der Controller- bereits auf die Platine gelötet ist, sollte der Rest leicht aufzubauen sein. Bitte achtet peinlichst darauf, daß alle Bauteile flach auf der Platine aufliegen!

Zunächst werden die beiden Zener-Dioden eingelötet. Man achte darauf, daß die Beinchen aller Bauteile direkt am Gehäuse gebogen werden. Am besten geht dies mit einer stabileren Pinzette. Die Zener-Dioden müssen entsprechend der Markierung richtig gepolt sein. Der Bestückungsdruck auf der Platine zeigt die richtige Anordnung, also mit dem schwarzen Balken der Diode zum USB-Stecker zeigend.

Dann folgen die sechs Widerstände. Diese haben keine Polung und können beliebig oder nach dem persönlichen, ästhetischen Empfinden eingelötet werden. Erwähnenswert sei noch die Tatsache, daß der 4,7K auch ein 10K sein kann oder umgekehrt! Die 1K Widerstände sind die Vorwiderstände für die beiden LEDs und können ggf. auch angepaßt werden.

Nach den Widerständen ist der kleine Keramik-Kondensator an der Reihe. Er trägt die Aufschrift "104", was für 100nF steht. Die Polung dieses 100nF ist beliebig.

Beim Quarz gibt es ebenfalls keine Polung. Man sollte aber vor dem Einlöten sicherheitshalber ein Stück Papier (doppelt gefaltet) zwischen die Beinchen des Quarzes legen, ihn dann festlöten und das Papier wieder herausziehen. Hintergrund für diese Maßnahme ist eine kleine Durchkontaktierung, die unter ungünstigen Umständen Kontakt zum Quarz bekommen könnte. Um dem vorzubeugen, lasse ich einfach diesen halben Millimeter Abstand zur Platine. Größere Abstände sollten allerdings nicht sein, damit der Quarz fest genug sitzt, nicht zu hoch wird und der Controller ordentlich getaktet werden kann.

So langsam nimmt die Platine Formen an. Als nächstes sind die beiden LEDs dran. Sie zeigen später die Versorgungsspannung und das Schreiben der Daten an. Das lange Beinchen einer LED ist immer plus und muß nach rechts zur Stiftleiste zeigen.

Zum Quarz gesellen sich nun noch die beiden Scheibenkondensatoren mit der Aufschrift "33CH". Das entspricht 33pF und sorgt dafür, daß der Quarz schön gleichmäßig schwingen kann.

Nun ist der richtige Zeitpunkt gekommen, um auch den Elko auf die Platine zu bringen. Da es sich um einen SMD-Elko handelt, ist die vorgehensweise etwas anders als bei den üblichen Bauteilen. Zuerst sollte man auf ein Lötpad (nicht auf beide) etwas Lötzinn geben. Nun legt man den Elko richtig gepolt über beide Löt pads und erhitzt das eben verzinnte Pad. Dabei schmilzt das Lötzinn, findet Kontakt zum Elko und ist damit schon mal fixiert. Achtet beim Elko auf die richtige Polung, sonst könnte es später qualmen. Der Minuspol des Elkos ist schwarz markiert und nicht zu übersehen!

Da nun der erste Kontakt hergestellt ist, kann das zweite Beinchen des Elkos ganz leicht festgelötet werden. Es reicht aus, wenn nur das zweite Löt pad erhitzt wird und man das Zinn in einer kleinen Menge zufließen läßt. Wer das langsam genug macht, wird erkennen, wie das Beinchen des Elkos vom Zinn umflossen wird. Damit ist ausreichend Kontakt hergestellt und der Elkos sitzt fest auf der Oberseite der Platine. - Gratuliere! Das war Dein erstes SMD-Bauteil.

Nachdem nun auch diese Hürde genommen ist, können die abgewinkelte Stiftleiste und der USB-Stecker angelötet werden. Da alle Bauteile von einer Seite der Platine bestückt werden, dürfte das keine weiteren Probleme machen.

Zum Schluß kommen noch die drei Stiftleisten für die Jumper. Damit später alles in das kleine Gehäuse paßt, empfehle ich die Stiftleisten mit der kurzen Seite nach oben einzulöten. Zwar sind die Kontakte dann nicht mehr sooo gut mit den Jumpers zu besetzen, aber das braucht man sowieso sehr selten oder gar nicht. Also scheut Euch nicht, die kurze Seite nach oben zu nehmen und den Überstand auf der Unterseite abzuschneiden.

Das war's auch schon mit dem Löten. Bitte überprüfe noch einmal anhand der Fotos, ob auch wirklich alle Bauteile richtig gepolt sind. Das ist sehr wichtig, damit später keine Schäden am ISP-Stick oder gar der USB-Buchse an Deinem Computer entstehen!

Funktionsprüfung und Installation

Der erste Funktionstest ist auch gleichzeitig mit der Installation des USB-Treibers verbunden. Sobald Du den ISP-Stick -entweder mit einem Kabel oder direkt- mit Deinem PC verbindest, leuchtet eine LED am ISP-Stick auf. Das ist schon mal ein gutes Zeichen, weil dann auch der Controller wirklich programmiert wurde. ;-)

Als nächstes meldet sich Windows, daß es ein neues USB-Gerät gefunden hat. Um den Treiber zu installieren, brauchst Du nur den Anweisungen auf dem Bildschirm folgen. Wird nach dem Treiber gefragt, wählst Du die Datei *usbasp.inf* oder *ispstick.inf*. Unter diesem Namen "usbasp" oder „ISP-Stick“ wird später der Programmierer im Gerätemanager auch angezeigt.

Die Kommunikation mit dem McCoi wird nun einfach über den 6-poligen Stecker hergestellt. Achtet unbedingt darauf, daß Ihr niemals den Jumper "V int" setzt, wenn der McCoi selbst am Strom hängt! Vorzugsweise läßt man alle Jumper vom ISP-Stick weg und versorgt den McCoi mit einer separaten Spannungsquelle. Das ist der sicherste Weg.

Ist die mechanische Verbindung hergestellt, kann man den McCoi nun mit dem Konfig-Tool programmieren. Das Konfig-Tool arbeitet im Hintergrund mit AVRdude (ab Version 5.2), was ideal für die Programmierung mit Hilfe des ISP-Stick ist.

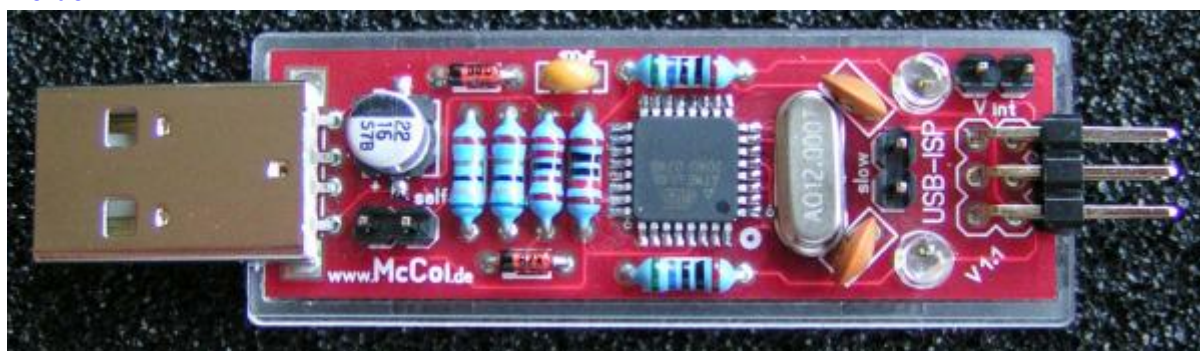
Der HEX-Schalter des McCoi muß für die Programmierung auf HEX-0 stehen!

target doesn't answer

Sollte die o.g. Fehlermeldung beim Konfig-Tool Version 1.4 auftauchen, hilft es, wenn man am ISP-Stick einmalig den Jumper bei "slow" setzt. Dadurch wird die Taktfrequenz dramatisch reduziert und der McCoi-Controller läßt sich programmieren. Allerdings dauert es eine halbe Ewigkeit, bis der Controller alles geladen hat.

Da ich (noch) nicht weiß, ob mit dieser langsamen Programmier-Art alle Parameter richtig geschrieben werden, empfehle ich die Programmierung auf normalem Wege (also OHNE Jumper) zu wiederholen! - Hierbei ist es sicherer, wenn alle Verbindungen zunächst gekappt werden und das Konfig-Tool neu gestartet wird. Zahlreiche Tests mit Controllern, die nicht auf Anhieb wollten, haben das bestätigt!

Übrigens ist es ganz gleich, wie Du das Kabel zwischen ISP-Stick und McCoi-Platine verbindest. Entweder klappt es oder es klappt nicht. Aber kaputt machen kannst Du nichts, solange Du keine Jumper gesetzt hast. - Also einfach ausprobieren. **Wenn Du den McCoi nur auslesen möchtest, mußt Du im KonfigTool zunächst eine Datei laden. Erst dann kann der McCoi ausgelesen werden!**

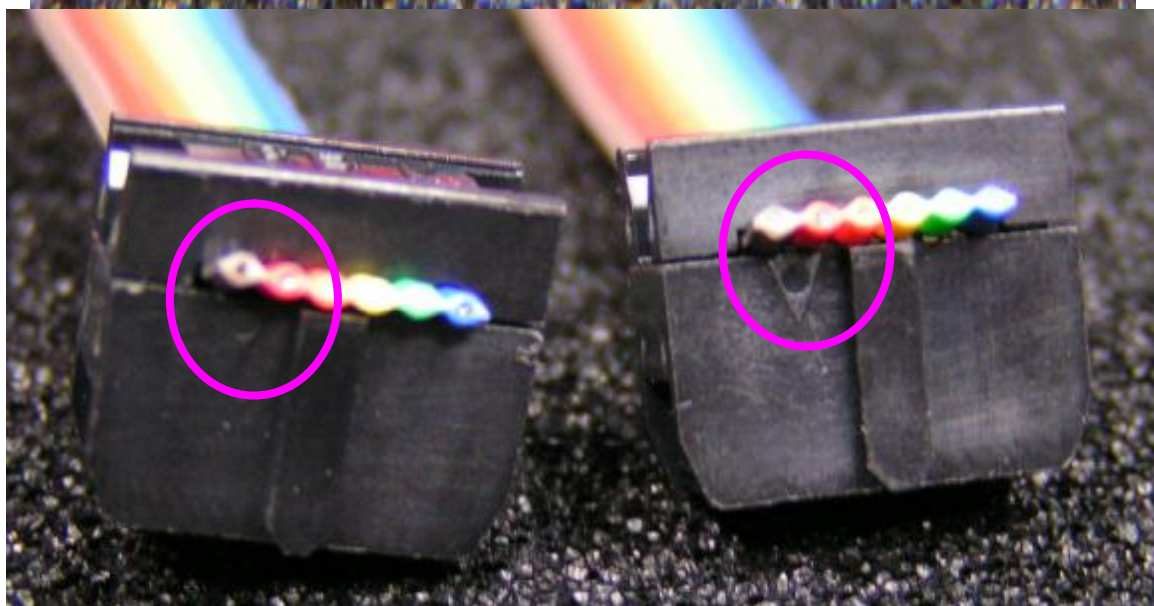
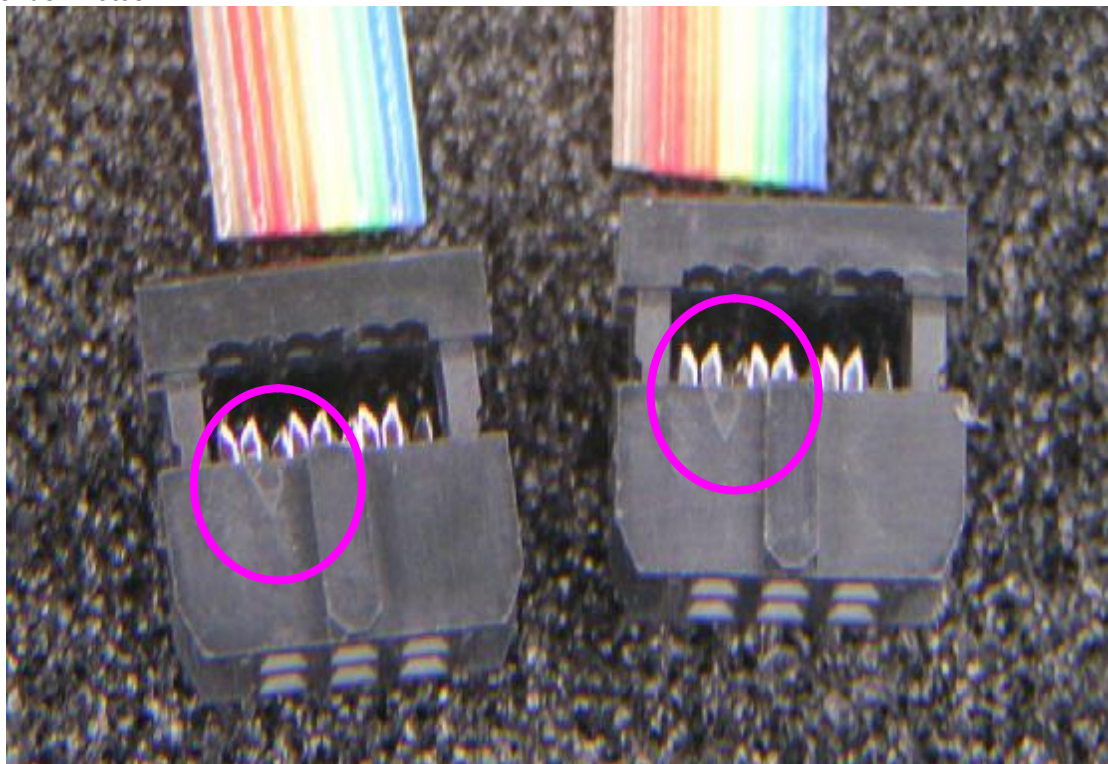


Ein kleiner Bug hat sich im KonfigTool leider auch eingeschlichen: Wenn Du statt des Cross-Schalters den Cross-Taster programmieren möchtest, funktioniert das zwar, aber beim erneuten Auslesen wird dieser Parameter immer wieder auf „Schalter“ gesetzt. Dies bedeutet NICHT, dass der Taster nicht programmierbar ist, sondern nur, dass das Auslesen an dieser einen Stelle einen Fehler macht. Funktionieren wird dennoch der Taster. Du kannst es dann ja einfach ausprobieren, indem du die Anschlüsse für Schalter oder Taster an der McCoi-Platine kurz überbrückst. Leuchtet die Cross-LED nach dem kurzen Überbrücken dauerhaft, ist der Taster programmiert. Leuchtet die Cross-LED nur so lange Du die Anschlüsse überbrückst, ist der Schalter programmiert.

Nochmals: Es besteht nur beim Auslesen der Fehler, nicht aber beim Schreiben der Daten in den Controller! Und dieses Problem betrifft nur den CrossModus, sonst keine anderen Funktionen. Beim nächsten Software-update des KonfigTools wird dieser Fehler behoben,... ☺

Nachtrag zum Regenbogenkabel.

Die kleinen Stecker des Regenbogenkabels haben ein kleines Dreieck als Markierung (links neben dem Steg). Achte darauf, dass die Kabelfarbe auf beiden Seiten gleich ist. Am besten zeigen das die folgenden Fotos:



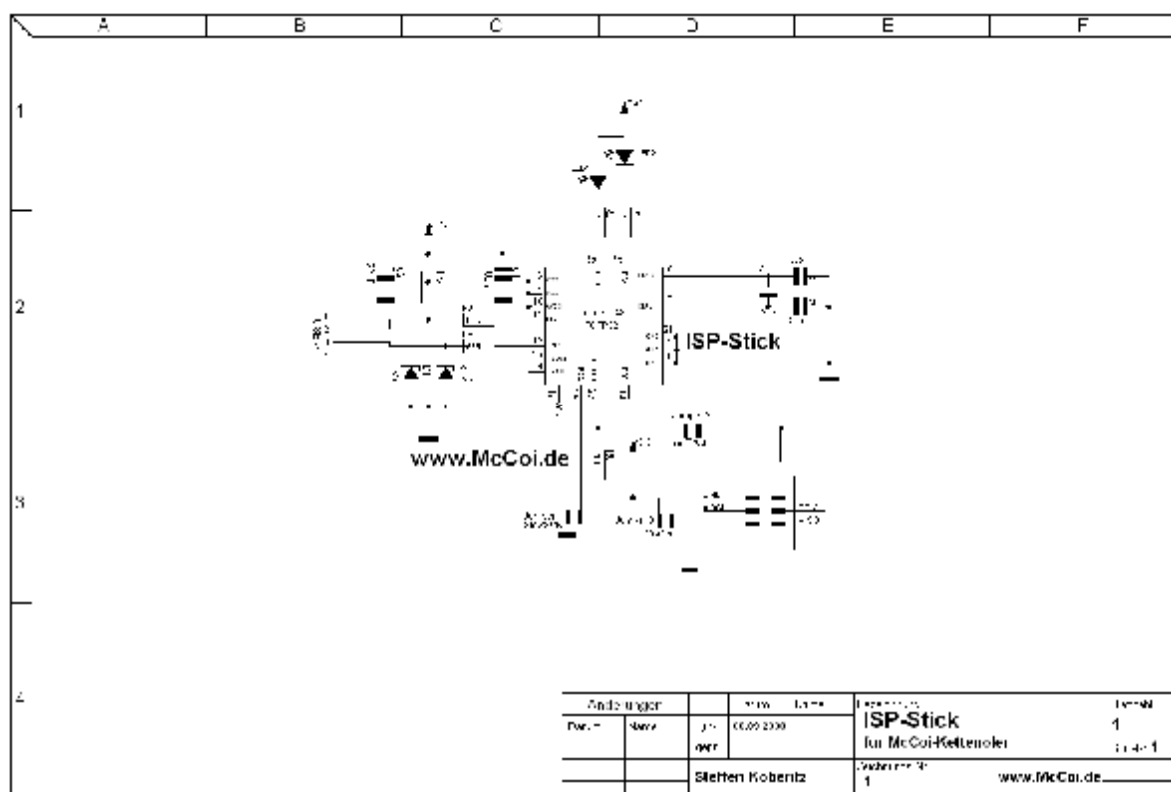
Lizenz

Dieses Projekt wurde unter GNU GPL version 2 realisiert. Mein Dank geht in erster Linie an die Firma OBJECTIVE DEVELOPMENT GmbH, die mit Ihrem USB-Treiber dieses Projekt möglich gemacht hat. Ebenso danke ich Thomas Fischl, der durch seine Vorlagen und die Anwendung des USB-Treibers Pate bei meinem ISP-Stick gestanden hat. Natürlich danke ich auch für die freundliche Genehmigung, einen Teil seiner Texte dem Projekt beifügen zu dürfen, um den Lizenzbestimmungen zu entsprechen. Weitere Informationen seien hier aufgeführt:

Links

[Firmware-only AVR USB driver](#)
[libusb](#)
[LibUsb-Win32](#)
[AVRDUDE - AVR Downloader/UploaDEr](#)
[Thomas Fischl](#)

Stromlaufplan



Bei Fragen, Wünschen, Bugs könnt Ihr jederzeit im [Forum](#) Hilfe finden oder mir eine [eMail](#) schreiben.

Haftung, EMV-Konformität

Der Nachbau und insbesondere die Inbetriebnahme, sowie Sicherheits- und EMV-Vorschriften unterliegen der Verantwortung desjenigen, der den Bausatz zusammenbaut und am Fahrzeug montiert. Ich übernehme keinerlei Haftung für evtl. auftretende Schäden oder Folgeschäden, die durch den Nachbau und die Inbetriebnahme direkt oder indirekt entstehen. Dies gilt gleichermaßen für Personen- Sach- und Vermögensschäden.

Fragen, Support und Bugs

Wenn Fragen auftauchen, bitte erst die Dokumentationen auf meiner Website (www.mccoi.de) gründlich lesen, das Forum befragen, die Schaltung überprüfen und mit den Mustern im Internet vergleichen. Ebenso sind die Erfahrungsberichte anderer McCoi-User sehr hilfreich! Wenn das alles nichts nutzt, bin ich im Forum oder per eMail unter frage@mccoi.de zu erreichen.