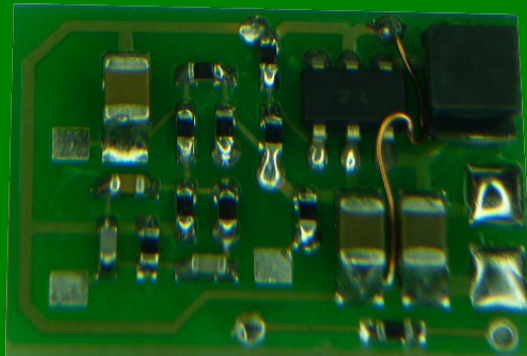
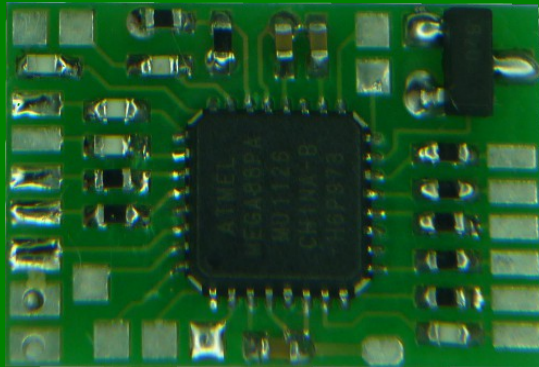


ENTWURF !!!

Open Car Decoder

Handbuch

Der Open Source DCC Car Decoder
unter General Public License
entworfen und entwickelt
von Toralf Wilhelm



Toralf Wilhelm
www.opcar.de

Version 1
April 2013

Rechtliche Hinweise:

Diese Anleitung wurde sorgfältig geprüft und nach bestem Wissen erstellt. Für die hier dargebotenen Informationen wird kein Anspruch auf Vollständigkeit, Aktualität, Qualität und Richtigkeit erhoben. Es kann keine Verantwortung für Schäden übernommen werden, die durch das Vertrauen auf die Inhalte dieser Anleitung oder deren Gebrauch entstehen.

Für den Inhalt und die Meinungen aller verlinkter Webseiten, tragen deren Betreiber Verantwortung. Sie wurden vor der Aufnahme auf rechts verletzende Inhalte geprüft. Sollten nachträglich rechtliche Beanstandungen zu finden sein, bitte ich um kurze Information, die Links werden dann umgehend entfernt. Alle hier genannten und gegebenenfalls durch Dritte geschützten Marken- und Warenzeichen sowie Patentrechte unterliegen uneingeschränkt den Bestimmungen des jeweils gültigen Rechts und den Besitzrechten der jeweiligen eingetragenen Eigentümer. Allein aufgrund der bloßen Nennung ist nicht der Schluss zu ziehen, dass Markenzeichen sowie Patente nicht durch Rechte Dritter geschützt sind.

Verbesserungsvorschläge und Hinweise auf Fehler bitte an:
mail@toralfwilhelm.de

Vielen Dank!

Inhaltsverzeichnis

EINLEITUNG.....	5
1 DECODERVERSIONEN.....	6
2 GRUNDSÄTZLICHES.....	7
2.1 Stromversorgung.....	7
2.2 Ansteuerung.....	8
2.3 Abstandsregelung / Rückmeldungen.....	8
3 ANSCHLÜSSE.....	9
3.1 IR Empfänger.....	9
3.2 IR Fototransistoren in der Fahrzeugfront.....	9
3.3 IR Sendedioden am Fahrzeugheck.....	9
3.4 Motoranschluss.....	10
3.5 LED Spannung.....	10
3.6 LED Ausgänge.....	10
3.7 Eingänge.....	11
3.8 Sonstige Anschlüsse.....	12
3.9 Anschlussplan.....	12
4 DECODERKONFIGURATION.....	14
4.1 Übersicht der CV Variablen.....	14
4.2 Reihenfolge der Decoderprogrammierung.....	15
4.2.1 Motor PWM Frequenz CV9.....	16
4.2.2 minimale Geschwindigkeit CV2.....	17
4.2.3 maximale Geschwindigkeit CV5 und CV23.....	17
4.2.4 Kickstart Spannung und Impulszeit CV66 und CV65.....	18
4.2.5 Fahrzeuggeschwindigkeit bei Fahrstufe 28 („volle Pulle“) in CV57.....	18

4.2.6 Beschleunigung CV3.....	19
4.2.7 Bremsen CV4 und CV56.....	19
4.2.8 Akkuspannung CV24 und CV25.....	19
4.2.9 Decoderadresse CV1 bzw. CV17 und CV18 und CV29.....	20
4.2.10 Fahrzeugtyp CV49.....	20
4.2.11 Konfiguration aller AUX CV42 und CV70 bis CV 118.....	20
4.2.12 Softwareupdate CV55.....	22
ANHANG.....	24

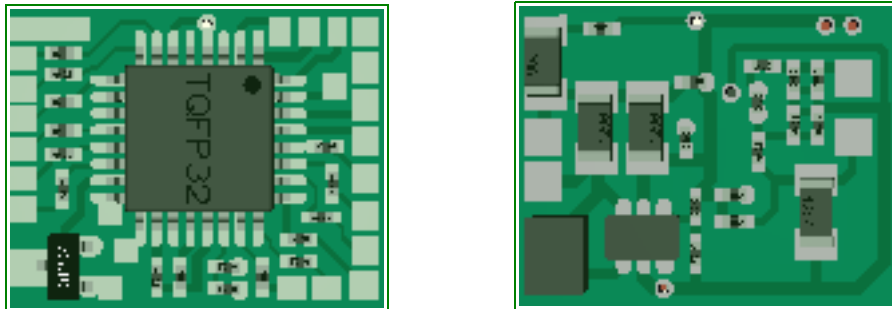
Einleitung

Der Open Car Decoder dient der Steuerung von Straßen Fahrzeugen auf der Modelleisenbahn. Er besteht hauptsächlich aus einem kleinen Mikrocontroller, welcher Steuerinformationen empfängt, auswertet und die gewünschten Funktionen setzt. Diese Steuerinformationen basieren auf normalen DCC Signalen und können von einer vorhandenen DCC digital Zentrale, einer PC Steuersoftware oder auch eigenständig von Lenz / Roco Lokmäusen erzeugt werden. Sie können auf verschiedenen Wegen zum Decoder übertragen werden. Der Car Decoder ist ein Teil des Open Car System, welches aktuell aus dem IR Booster (Steuerinformation Sender), dem CV – Prog (Car Decoder Programmieradapter) und dem Steuer- und Rückmelde Modul besteht. Dieses Dokument erläutert die Aufbaumöglichkeiten eines Open Car System Fahrzeug unter Verwendung eines Open Car Decoder und dessen Ausgangsprogrammierung im Systemverbund Open Car System.

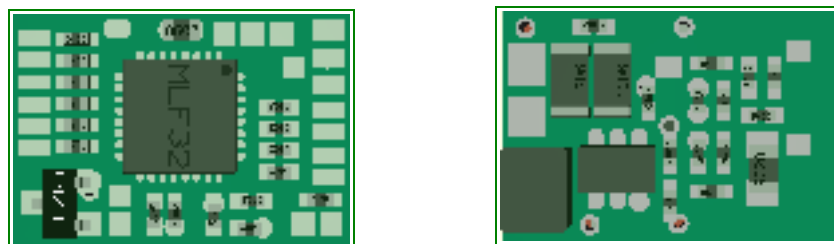
Vor dem Einbau der Decoder, sollte sich jeder über die Grundsätzliche Konfiguration seiner Fahrzeuge Gedanken machen! Dazu gehört, Akkutyp, Ladetechnik, Position und Typ der Bauteile für die Abstandsregelung, Fahrzeuggeschwindigkeiten, Anschlussart des Fahrmotors (und damit Konfiguration der Abstandsregelung), Verwendung der originalen Stopfstelle.

1 | Decoderversionen

Zum Systemstart gibt es die Decoder in der Hardware Version 1 in zwei unterschiedlichen Baugrößen. Mit 14x18 mm ist dies die größere Variante:



und hier nicht Maßstabgerecht ! die kleine mit 10x14 mm:



Beide Decoder der Version 1 gibt es nur als nackte Platinen zum selber bestücken. In überschaubaren Mengen, stelle ich auf Anfrage per Hand bestückte Muster (für den aktuellen Reichelt Materialpreis + 1,50 € Platine + 5,00 Bestückungspauschale + Porto) zur Verfügung. Für den Anwender unterscheiden sich die beiden Baugrößen nur marginal, so das die Anleitung für Version 1 identisch ist.

Die nächste Hardwareversion wird es voraussichtlich auch als fertig bestückte Platinen geben, wir wollen auf jeden Fall an zwei Baugrößen festhalten, da die kleine Version so filigran ist, das schon das anlöten der Verkabelung für die Fahrzeugbeleuchtung für die meisten Anwender unüberwindbare Hürden darstellt. Es wird also auch Zukünftig eine „größere“ Version geben, welche sich leichter einbauen lässt. Was in großen Fahrzeugen (LKW, Bus) sicherlich kein Nachteil ist. Diesen wird es als Platine und als fertig aufgebauten Decoder geben. Die kleinere Version, wird es dann nur noch fertig Aufgebaut geben.

2 | Grundsätzliches

2.1 | Stromversorgung

Die Fahrzeugdecoder werden üblicher weise aus Akkus versorgt. Dabei können die Decoder mit und ohne integrierten Spannungswandler aufgebaut werden. Standard Empfehlung ist der Aufbau mit Spannungswandler. Die Spannungswandler sorgen für eine konstante Decoderversorgungsspannung von ca. 4,3V. Was auch den Betrieb von weißen und blauen LEDs an 1,2V und 2,4V Akkus ermöglicht. Der Spannungswandler arbeitet ab einer Eingangsspannung von 0,9V. Wobei der maximal zur Verfügung stehende Ausgangsstrom von der Eingangsspannung abhängig ist! Als Richtwert gilt folgendes:

minimale Eingangsspannung	maximaler Ausgangsstrom
0,9V	80mA
2,0V	200mA
3,5V	350mA

Dabei benötigt der Decoder selbst, (eingeschaltet mit Rückmeldung in Bereitschaft) ca. 20mA. Wobei das meiste davon die IR Dioden für die Abstandsregelung / Rückmeldung verbrauchen. Beim Akku, liegt die Empfehlung eindeutig bei einer LiIonen / LiPolymer Zelle. Diese sind inzwischen preiswert erhältlich, im Verhältnis klein und leicht und mit 3,3 – 4,2 V Akkuspannung optimal geeignet.

An dieser Stelle, schon eine Bemerkung im Zusammenhang mit der Abstandsregelung zwischen den Fahrzeugen. Um ein möglichst ruhiges und exaktes regeln des Fahrzeugabstand zu ermöglichen, ist es günstig die Geschwindigkeit des voraus fahrenden Fahrzeug zu kennen. Diese übertragen die Fahrzeuge mit dem Rückmeldesignal. Das ist allerdings ein errechneter Wert (die Fahrzeuge haben ja keinen Tacho) der hauptsächlich von der Motorspannung beeinflusst wird. Aus diesem Grund ist es günstig, den Fahrzeugmotor auch mit an der stabilen Decoderspannung von 4,3V zu betreiben. Was wiederum voraussetzt, das dieser vorhanden ist (auch bei Li-Zellen mit 3,3 – 4,2V). Es besteht zwar die Möglichkeit, die Motoransteuerung und somit die Fahrzeuggeschwindigkeit in Abhängigkeit der Akkuspannung anzupassen, dies muss aber per CV an jedes Fahrzeug einzeln angepasst werden und verringert den möglichen Geschwindigkeitsregelbereich um den Bereich für die Akkuanpassung.

Zusammenfassung, optimale Decoderversorgung: eine Li-Zelle, Platine mit Spannungswandler bestückt, Motor über stabile Decoderspannung versorgen.

2.2 | Ansteuerung

Möglichkeit 1: moduliertes IR DCC Signal über mehrere Meter mit 28 Fahrstufen, DCC Funktionen F0 – F8, lange und kurze DCC Adressen möglich.

Möglichkeit 2: unmoduliertes IR DCC Signal bis ca. 30cm mit 28 Fahrstufen, DCC Funktionen F0 – F8, lange und kurze DCC Adressen möglich, Service Mode Programmierbefehle mit Rückmeldung über Frontscheinwerfer.

Möglichkeit 3 (in Planung): 2,4 GHz Funksteuerung als alternative zum modulierten IR DCC Signal

2.3 | Abstandsreglung / Rückmeldungen

Die Fahrzeuge senden über IR Dioden im Fahrzeugheck folgende Informationen:

- 16 Bit Fahrzeugadresse
- aktuelle Geschwindigkeit in cm/Sek
- Fahrzeugtyp
- Akkuladestand in %
- Beleuchtungsstatus für Anhänger

Diese Information können von Rückmeldedecoder sowie auch einem folgendem Fahrzeug empfangen werden und zu Rückmeldezwecken bzw. zur Abstandsreglung zwischen den Fahrzeugen benutzt werden. Die Rückmeldesignale werden im DCC Format mit zwei unterschiedlichen IR Strahlungsintensitäten am Fahrzeugheck abgestrahlt. Die Abstandsreglung im folgendem Fahrzeug erkennt die Signale dadurch in zwei unterschiedlichen Entfernungen, so das z.B. auf ein stehendes Fahrzeug dichter aufgefahren werden kann, als auf ein fahrendes.

Die Abstandsreglung kann über die Funktion F4 abgeschaltet werden, um z.B. an einem stehenden Fahrzeug vorbei fahren zu können.

3 | Anschlüsse

3.1 | IR Empfänger

Zum Empfang der 455kHz modulierten DCC Steuerinformation wird ein externer IR Empfänger benötigt. Dieses Bauteil (TSOP7000) wird an die Decoder Anschlüsse +UB TSOP, GND TSOP, IR TSOP angeschlossen. Wobei zu beachten ist, das bei der kleinen Version 1 Decoderplatine auch der 4,7uF Kondensator C10 zwischen +UB TSOP und GND TSOP extern angeschlossen werden muss.

3.2 | IR Fototransistoren in der Fahrzeugfront

Zum einlesen der Informationen für die Abstandsreglung und zur Kurzstreckensteuerung mit unmodulierten DCC - IR Signalen wird an der Fahrzeugfront mindestens ein besser zwei (1x vorn rechts, 1x links) Fototransistoren benötigt. Wenn zwei oder mehr Transistoren verwendet werden, müssen diese parallel geschaltet werden. Dabei werden die Kollektoren mit Foto+ und die Emittoren mit Foto- verbunden. Optisch hat es sich als günstig erwiesen, wenn die Transistoren je etwas nach außen gerichtet (also nicht gerade nach vorn) angebracht werden. In Kurvenfahrten ist so der Vordermann in einem größeren Abstand wahrnehmbar. Die Transistoren können auch vorn unter dem Fahrzeug angebracht werden (das ist Optisch hübscher), wichtig ist nur, das sie freie Sicht nach vorn haben. Sollten die Fototransistoren empfindlich auf normales Umgebungslicht bzw. Sonnenlicht reagieren (was die Abstandsreglung deutlich stört), können sie komplett (auch vorn) mit einem schwarzem „Edding“ angemalt werden. Das dämpft das IR Abstandssignal nur marginal, blockt aber normales Licht erfolgreich.

3.3 | IR Sendedioden am Fahrzeugheck

Es werden zwei IR Dioden in Reihe (Polarität beachten!) benötigt. Diese sollten am Fahrzeugheck rechts und links angebracht werden. Im Gegensatz zu den Fronttransistoren können diese gerade nach hinten gerichtet angebracht werden, auch sie dürfen unter dem Fahrzeug angebracht sein, Hauptsache freie Sicht nach hinten. Die Dioden werden vom Anschluss +UB 4,3V (Anode Diode1) zu den Anschlüssen IR_TX und IR_X_low (Kathode Diode 2) angeschlossen. Bei Platinen Version 1 gibt es noch beide, ab Version 2 gibt es nur noch IR_TX.

3.4 | Motoranschluss

Für den Fahrzeugmotor gibt es zwei Anschlussmöglichkeiten. Bei einer Akkuspannung ab 2,4V (zwei NiCd bzw. NiMh Zellen) empfehle ich den Fahrmotor mit an den Anschluss für die stabile +UB 4,3V LED Spannung und dem Motor Anschluss anzuschließen. Bei Betrieb an nur einer Zelle mit 1,2V ist dies nur bei sehr kleinen Motoren möglich (siehe Stromversorgung). Diese Anschlussart ist vorzuziehen und es ist zum Schutz der Decoderbauteile eine externe Schutzdiode (1N4148 reicht aus) über dem Motoranschluss anzubringen. Ist diese Anschlussversion nicht möglich, muss der Fahrmotor zwischen Akku+ und dem Motor Anschluss angeschlossen werden. In diesem Fall sollte in CV 23 ein Ausgleich für die Motoransteuerung bei sinkender Akkuspannung programmiert werden, um eine gleichmäßige Funktion der Abstandsregelung zu gewährleisten. Sollte ein Originalmotor zu schnell laufen, da er ja am Decoder unter Umständen mit einer erheblich höheren Spannung läuft, sollte man 1 bis 3 Dioden (1N4148 sollten reichen) in Reihe in die Motorversorgungsleitung einbauen. Die maximale Motorgeschwindigkeit lässt sich in CV5 zwar begrenzen, aber es fließen dann kurzzeitig sehr hohe Motorströme, was sowohl den Spannungsregler wie auch den Akku ungünstig belastet und auch den maximalen Regelbereich von CV5 unnütz einschränkt.

3.5 | LED Spannung

Ist der Decoder mit einem StepUp Spannungswandler ausgestattet (das ist die Empfehlung) steht am Anschluss „+UB 4,3V“ eine konstante Spannung von 4,3V für die externen Komponenten zur Verfügung. Diese kann je nach Eingangsspannung (siehe Betriebsspannung) mit bis zu 350mA belastet werden. Hierüber können alle LEDs / IR Dioden und meist auch der Fahrmotor betrieben werden. Das hat den Vorteil, das es keine Helligkeitsschwankungen bzw. Geschwindigkeitsschwankungen bei sinkender Akkuspannung gibt.

3.6 | LED Ausgänge

Der Decoder verfügt über zehn LED Ausgänge, die über die Funktionen F0 bis F8 schaltbar sind. Wobei F3 und F4 sonder Funktionen haben. Alle LED Ausgänge besitzen Vorwiderstände auf der Platine, welche für weiße und blaue LEDs 220 Ohm und für rote, orange, gelbe LEDs 470 Ohm betragen. Ausnahme ist das Rücklicht, hier sind 1000 Ohm vorgesehen. Alle LED Ausgänge schalten Standardmäßig nur nach GND (minus). Das bedeutet, wenn der Ausgang eingeschaltet ist, hat er Verbindung nach GND, wenn er ausgeschaltet ist, ist er hochohmig. Es dürfen aber trotzdem keine externen Spannungen an die Ausgänge angelegt werden! Alle Verbraucher dürfen maximal an die

„+UB 4,3V“ angeschlossen sein (im Normalfall gibt es auch keine anderen Spannungen im Fahrzeug ich erwähne das hier nur zur Sicherheit). Es ist allerdings zulässig, LED Ausgänge zu verbinden, um eine LED mit verschiedenen Funktionen gleichzeitig zu betreiben. Die Kombination aus Rücklicht und Bremslicht ist so ein Beispiel, man benötigt dann nur eine rote LED (auf jeder Seite eine) am Fahrzeugheck und diese wird bei eingeschaltetem Locht (F0 ein) über den 1000 Ohm Vorwiderstand eingeschaltet und zusätzlich noch beim bremsen über den 470 Ohm Widerstand viel heller als Bremslicht verwendet (Vorteil man erspart sich 2 Mini LEDs zu verkabeln).

Eine Ausnahme bildet der Anschluss Licht_3/AUX_7/geschaltet mit F7. Dieser kann über CV42 mit verschiedenen Ausgangspegeln (Schaltverhalten) programmiert werden, um auch einen Ausgang zu haben, welcher nach „plus“ schaltet, um z.B. externe Bauteile / Transistoren damit direkt schalten zu können. Sollte dieser Ausgang auf „plus“ umprogrammiert sein, darf er nicht mehr mit einem anderen AUX extern verbunden werden! Die Zuordnung der LED Ausgänge zu den Funktionen ist folgende:

F0	AUX_1 / Frontlicht und AUX_2 / Schlusslicht
automatisch	AUX_3 / Bremslicht
F1	AUX_4 / Blinker links
F2	AUX_5 / Blinker rechts
F5	AUX_9 / Rundumlicht 1 und AUX_10 / Rundumlicht 2
F6	AUX_6 / Licht 2
F7	AUX_7 / Licht 3
F8	AUX_8 / Licht 4

Die Lichteffekte der Ausgänge AUX_6 bis AUX_10 (F5-F8) lassen sich über die CV 70 – 118 frei mit Effekten programmieren (sieh bei CV Beschreibungen).

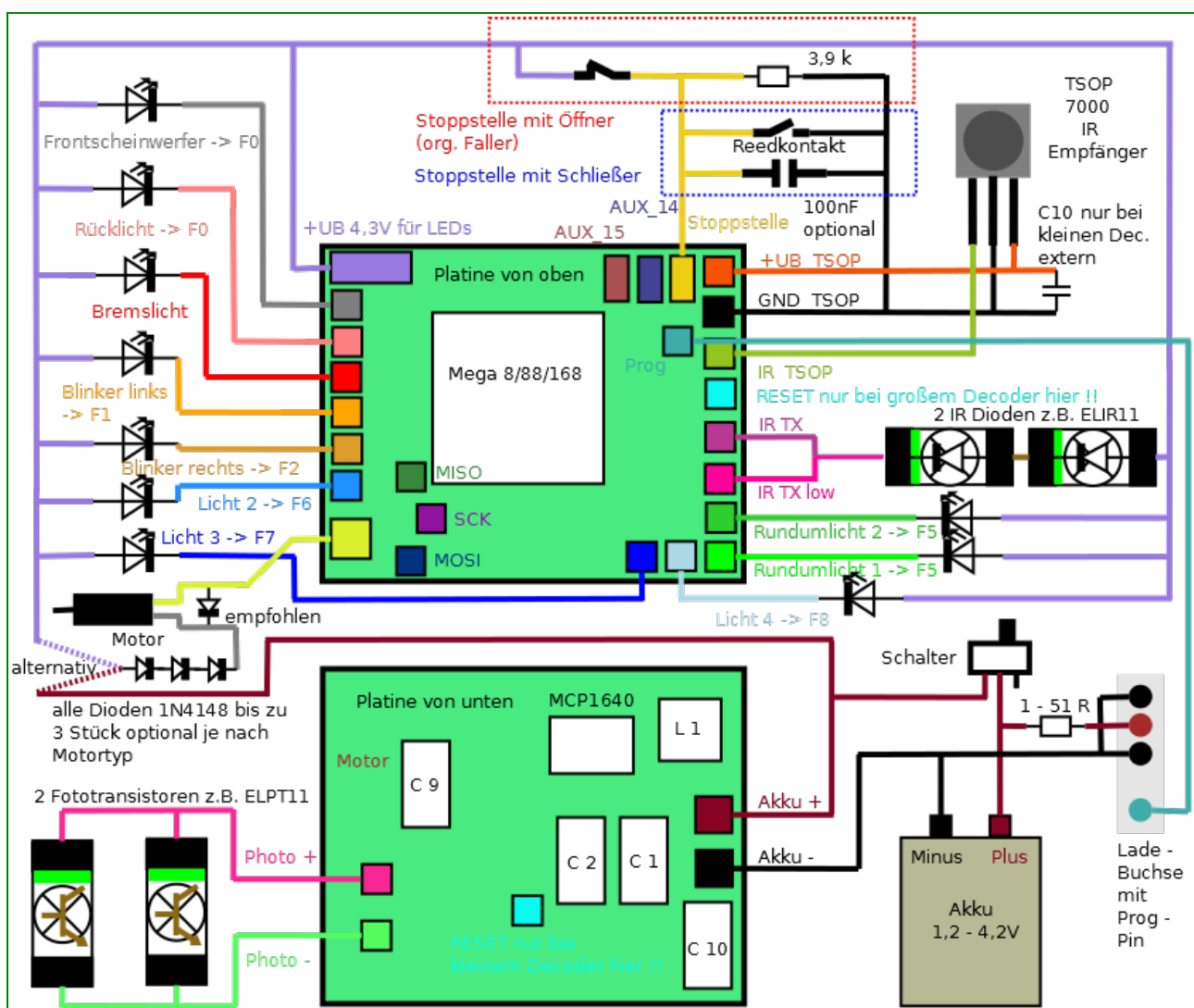
3.7 | Eingänge

Es sind drei Funktionseingänge vorhanden (AUX_13 bis AUX_15). Wobei aktuell nur AUX_13 für die Original Stoppstelle benutzt wird. Dabei ist zu beachten, das AUX_13 low aktiv ist. Das bedeutet, Stopp wird bei low Pegel (0V) am AUX_13 erkannt! Die Funktion der Stoppstelle kann mit F3 abgeschaltet werden, so das man Stoppstellen bewusst überfahren kann. AUX_14 und AUX_15 sind aktuell noch unbenutzt.

3.8 | Sonstige Anschlüsse

Akku+ und Akku- sind die Anschlüsse für den Akku (wie der Name schon sagt). Der Anschluss „Prog“ sollte nach Möglichkeit von außen zugänglich sein, meine Empfehlung ist in der Nähe von der Ladebuchse. Über diesen Anschluss, lässt sich die Decoder Software austauschen. (siehe Software Update) . Die Anschlüsse MISO, MOSI, SCK und RESET werden zur Erstprogrammierung des Mikrocontroller benötigt. Im normalen Betrieb sind sie momentan ungenutzt.

3.9 | Anschlussplan



Hier zur Übersicht der komplette Anschlussplan, wobei für den Motoranschluss und die Stoppstelle optional beide Versionen eingezeichnet sind. Es wird natürlich immer nur eine Version angeschlossen! → siehe Text. Der RESET Anschluss ist auf dem kleinen Decoder auf der Unterseite! Bei der Belegung der Anschluss pads aufpassen, bei dem kleinen fehlt dieser auf der Oberseite!

4 | Decoderkonfiguration

4.1 | Übersicht der CV Variablen

CV	Standartwert	Beschreibung
1	3	Decoderadresse (kurze Adresse 1 – 127 wie DCC Norm)
2	10	Minimale Geschwindigkeit
3	10	Anfahrverzögerung
4	1	Bremsverzögerung
5	180	Maximale Geschwindigkeit (28 + CV2 bis 255)
7	xx	Softwareversion
8	13	Herstellerkennung
9	6	1 – 7 Motor PWM Frequenz 1 = 31kHz, 4kHz, 1kHz, 500Hz, 250Hz, 125Hz, 7 = 30Hz
17	192	Erweiterte Adresse Höherwertiges Byte der langen Adresse plus 192
18	3	Erweiterte Adresse Niederwertiges Byte der langen Adresse
23	0	Faktor zur Geschwindigkeitsanpassung bei sinkender Akkuanpassung 0 = aus bis 10 = maximale Anpassung
24	33	Minimale Akkuspannung (Akku leer Meldung kommt → Warnblinker ein) in V * 10 (Akkuspannung ohne Komma)
25	42	Maximale Akkuspannung (voller Akku) in V * 10 (Akkuspannung ohne Komma)
26	0	Geschwindigkeit (PWM Pulsbreite) nach Decoderstart 0 - 255
27	0	Reserviert für Oszillatorabgleich
29	0	Konfigurationsvariable nach DCC Norm +32 : erweiterte lange Adresse CV17/18 sonst CV1
42	0	AUX_7 Konfiguration +0 : AUX_7 aus → hochohmig ein → GND +16 : AUX_7 aus → +UB ein → GND +32 : AUX_7 aus → GND ein → +UB
49	1	Fahrzeugtyp zur freien Verwendung, wird rückgemeldet

CV	Standartwert	Beschreibung
55	0	Softwareupdate ungleich 0 → Decoder im Update Modus
56	1	Bremsverzögerung in der Abstandsreglung (CV4 bei ASR)
57	25	Fahrzeuggeschwindigkeit bei Fahrstufe 28 in cm / Sekunde
65	5	Kickstartzeit (CV65 * 8ms)
66	80	Kickstart PWM Pulsbreite (0 bis 255)
70 - 77	xx	Effektzeiten AUX_6 / F6 je 0 – 255 * 8ms 70 ein, 71 aus, 72 ein, 73 aus, 74 ein, 75 aus, 76 ein, 77 aus
78	3	Wiederholungen der Effekte von AUX_6 1 – 253 Anzahl der Wiederholungen, 254 ständig wiederholen, 255 AUX ständig an
80 - 87	xx	Effektzeiten AUX_7 / F7 (wie CV 70 – 77)
88	254	Wiederholungen der Effekte von AUX_7 (wie CV 78)
90 - 97	xx	Effektzeiten AUX_8 / F8 (wie CV 70 – 77)
98	254	Wiederholungen der Effekte von AUX_8 (wie CV 78)
100- 107	xx	Effektzeiten AUX_9 / F5 (wie CV 70 – 77)
108	254	Wiederholungen der Effekte von AUX_9 (wie CV 78)
110- 117	xx	Effektzeiten AUX_10 / F5 (wie CV 70 – 77)
118	254	Wiederholungen der Effekte von AUX_10 (wie CV 78)

4.2 | Reihenfolge der Decoderprogrammierung

Folgend eine Empfehlung zur Reihenfolge der Decoderprogrammierung:

1. Motor PWM Frequenz CV9
2. minimale Geschwindigkeit CV2
3. maximale Geschwindigkeit CV5 und Geschwindigkeitsanpassung bei sinkender Akkuspannung CV23 (nur bei Motor an Akku+ nötig)

4. Kickstart Spannung und Impulszeit CV66 und CV65
5. Fahrzeuggeschwindigkeit bei Fahrstufe 28 („volle Pulle“) in CV57
6. Beschleunigung CV3
7. Bremsen CV4 und CV56
8. minimale und maximale Akkuspannung CV24 und CV25
9. Decoderadresse CV1 bzw. CV17 und CV18 und CV29
10. Fahrzeugtyp CV49
11. Konfiguration aller AUX CV42 und CV70 bis CV 118

Es wird empfohlen, vor dem programmieren den Fahrzeugakku aufzuladen! Zum ermitteln des Wertes für CV57 bei Betrieb des Motors direkt am Akku ist die zwingend notwendig!

4.2.1 | Motor PWM Frequenz CV9

Der Fahrmotor wird mit einer Pulsbreitenmodulation angesteuert. Das bedeutet, wenn der Motor 50% seiner Maximalgeschwindigkeit fahren soll, wird der Motortreiber nicht etwa zur Hälfte geöffnet und es liegt so die halbe Betriebsspannung am Motor an (klassischer Analogbetrieb), sondern es wird die Endstufe für eine bestimmte Zeit komplett eingeschaltet (womit die volle Betriebsspannung am Motor anliegt) und danach die selbe Zeit komplett ausgeschaltet. Im Durchschnitt dieser Zeit (welche die PWM Frequenz ist) liegt dabei dann auch die halbe Betriebsspannung am Motor an und dieser dreht sich mit halber Maximalgeschwindigkeit. Kurz gesagt, der Motor wird ständig ein und ausgeschaltet. Je länger die Einschaltzeit ist, um so schneller dreht er sich, wobei die Gesamtzeit (aus ein und aus Zeit) immer gleich bleibt. Um so öfter man dieses wiederholt, um so höher ist die Frequenz, mit der der Motor angesteuert wird und um so ruhiger läuft er. Wobei es hier leider ein paar elektrische und mechanische Grenzen, je nach verwendeten Motortyp gibt. Des weiteren hört der Mensch leider bestimmte Frequenzen und findet dies unangenehm (pfeifen und piepen der Motoren). Bis 250Hz PWM Frequenz brummt der Motor, danach fängt er an zu pfeifen, was erst über der Hörschwelle von ungefähr 16kHz nicht mehr wahrnehmbar ist. Leider, lassen sich nicht alle Motoren mit einer hohen Frequenz von z.B. 32kHz betreiben, weil hier durch die Trägheit des Ankers (die Wicklungsinduktion), die Motoren bei kleinen Drehzahlen und somit sehr kurzen Ansteuerimpulsen kaum noch ein Drehmoment erreichen. Zusammengefasst, lässt sich sagen, um so mehr „Eisen“ im Anker vorhanden ist, um so geringer ist die optimale PWM Frequenz für den Motor. Im Umkehrschluss bedeutet das, ein Faulhaber - Motor lässt sich prima geräuschlos mit 32 kHz Motor PWM Frequenz ansteuern. Die originalen Faller Motoren laufen bei mir mit 125 – 250 kHz am besten. Da die PWM Frequenz wesentlich das Drehmoment der Motoren beim anfahren beein-

flusst, ist die zuerst vor CV2 Startgeschwindigkeit und CV5 Maximalgeschwindigkeit einzustellen. Die möglichen Werte für CV9 Motor PWM Frequenz sind:

1 = 32kHz, 2 = 4kHz, 3 = 1kHz, 4 = 500Hz, 5 = 250Hz, 6 = 125Hz, 7 = 30Hz

alle anderen Werte werden beim Decoderstart intern durch 6 = 125Hz ersetzt. Man kann hier ruhig einmal die verschiedenen Werte durchprobieren und sich seine Favoriten Einstellung suchen. Dabei ist es normal, das die Motorstartgeschwindigkeit nicht optimal ist (die stellen wir danach ein), bei hohen Frequenzen, kann es sein, das man denkt „der Motor fährt ja überhaupt nicht mehr“, erfährt aber auch da los, allerdings durch das geringe Drehmoment erst sehr spät.

4.2.2 | minimale Geschwindigkeit CV2

Nach der Motor PWM Frequenz kann man die davon beeinflussten Einstellungen für die minimale Motorspannung (und auch die weiteren) machen. Momentan noch nicht davon irritieren lassen, das der Motor eventuell einen kleinen „Sprung“ beim anfahren macht, das ist der „Kickstart“ den stellen wir noch ein. Jetzt erst einmal das Fahrzeug mit der kleinsten Fahrstufe laufen lassen und CV2 so einstellen, das es sicher noch fährt (auch bei unterschiedlicher Belastung von z.B. dem Lenkmagneten).

4.2.3 | maximale Geschwindigkeit CV5 und CV23

Als nächstes, wird die Maximalgeschwindigkeit des Fahrzeuges eingestellt. Diese lässt sich von dem Wert in CV2 (Startgeschwindigkeit) + 28 (wir fahren mit 28 Fahrstufen und jede sollte schon 1x möglich sein) und 255 (wir arbeiten intern mit 8 Bit) einstellen. Wobei folgendes zu beachten ist, wenn der Fahrzeugmotor vom Motoranschluss nach Akku+ angeschlossen ist, verändert sich die Betriebsspannung für den Motor mit sinkender Akkuspannung! Bei z.B. einem LiIonen Akku um fast 1V (Akku voll 4,2V Akku leer 3,3V), was ca. 25% ausmacht und deutlich im Fahrbetrieb erkennbar ist. Aus diesem Grund, gibt es die CV23. Hier lässt sich dafür ein Korrekturwert ablegen, mit welchem in Abhängigkeit der Akkuspannung die Motorgeschwindigkeit korrigiert werden kann. Bei den oben erwähnten 25% braucht dieser Korrekturwert die Möglichkeit diese 25% auszugleichen. Aus diesem Grund verändert sich der Regelbereich für CV5 in Abhängigkeit von CV23. Das wird intern berechnet und vom maximal möglichem Wert für CV5 von 255 intern abgezogen, so das der Maximalwert für CV5 sich dann intern verringert (man merkt dies nur an der größeren Abstufung zwischen den Fahrstufen). Wenn also CV23 verwendet werden soll, dann bitte zuerst CV23 programmieren, weil dadurch automatisch das Fahrzeug (mit vollem Akku) langsamer wird. Die Werte für CV23 dürfen zwischen 0 (Korrektur aus) und 10 (maximale Korrektur) liegen. Werte über 10 werden beim Decoderstart intern auf 10 begrenzt. Für den oben erwähnten LiIonen Akku liegt der Wert in CV23 bei ca. 8 . Abschließend sei noch bemerkt, das der Decoder intern mit

8 Bit Werten rechnet und die Korrektur der Geschwindigkeit durchaus dazu führen kann, das das Fahrzeug bei sinkender Akkuspannung und steigender Korrektur in den Übergängen zu Geschwindigkeitsschwankungen neigt. Dies ist auch ein Grund dafür, den Fahrmotor mit an der stabilen Decoderspannung von 4,3V zu betreiben. Notfalls mit den drei Dioden in der Motorzuleitung (siehe Anschlussplan). In diesem Fall bleibt CV23 \rightarrow 0 und es muss nur CV5 eingestellt werden.

Dieser gesamte Abgleich der Motorspannung ist nötig, um ein möglichst gleichmäßiges ausregeln des Fahrzeugabstand in der Abstandsreglung zu ermöglichen. Dafür ist es auch von Vorteil, wenn man alle seine Fahrzeuge auf eine annähernd gleiche Maximalgeschwindigkeit abgleicht. Das ist nicht zwingend Notwendig, ermöglicht aber ein gleichmäßiges Abbremsen des auffahrenden Fahrzeuges.

4.2.4 | Kickstart Spannung und Impulszeit CV66 und CV65

Diese beiden CV ermöglichen ein besseres andrehen des stehenden Fahrmotor. Wenn das Fahrzeug aus dem Stand losfährt, wird jedes mal für die Zeit in CV65 * 8ms der Fahrmotor mit der PWM Pulsbreite aus CV66 angesteuert.

4.2.5 | Fahrzeuggeschwindigkeit bei Fahrstufe 28 („volle Pulle“) in CV57

Da jedes Fahrzeug auch (bei der gleichem Fahrstufe) eine andere Geschwindigkeit hat, sollte hier die Fahrzeuggeschwindigkeit bei Fahrstufe 28 und vollem Akku in cm je Sekunde abgelegt werden. Dieser Wert muss nicht exakt ermittelt werden, man kann ihn auch einfach so verändern, das ein auffahrendes Fahrzeug nicht hinten in den Vordermann rein fährt, sondern seine Geschwindigkeit spätestens nahe hinter dem Vordermann deutlich unter die des Vordermann absenkt. Wichtig ist, das der Abgleich bei alle Fahrzeugen identisch durchgeführt wird und dann höchstens noch ein Feintuning erfolgt. Empfehlen kann ich folgende Vorgehensweise, eine feste Fahrstrecke vermessen z.B. 10 Meter. Wer nur einen kürzeren Testkreis hat, kann den auch mehrfach nehmen. Dann das Fahrzeug (mit vollem Akku) bei Fahrstufe 28 (V max.) diese Strecke fahren lassen und die Zeit stoppen, welche es dafür benötigt. Das RC1 Testcar benötigt für 10 m ca. 35 Sekunden. Jetzt also $10 / 35 = 0,28 \rightarrow$ RC1 fährt mit einer Maximalgeschwindigkeit von 28 cm / Sekunde. In CV57 wird also 28 geschrieben. Der Decoder begrenzt allerdings Werte von über 50 cm / Sekunde auf 50 cm / Sekunde (das ist trotzdem schon Modelleisenbahn – Formel 1 fahren). Mit diesem Wert kann der Decoder immer seine aktuelle Geschwindigkeit errechnen (er hat ja leider keinen Tacho) und diese nach hinten weitermelden. Mit diesem und seinem eigenem Wert kann ein nachfolgendes Fahrzeug seine Geschwindigkeit in der Abstandsreglung feiner anpassen, so das ein gleichmäßiges Regeln möglich ist. Bitte hier bei der Ermittlung des Wertes nicht übertreiben, es ist nur ein ungefähre Anhaltspunkt der Fahrzeuggeschwindigkeit für die Abstandsreglung nötig! Hier wird überall

intern auf 8 Bit gerundet. Es wird mit diesem Verfahren möglich, zwischen einem Trecker der mit Fahrstufe 28 vor einem her kraucht und einem PKW der mit Fahrstufe 28 rast zu unterscheiden. Die Rückmeldedecoder können so auch die aktuelle Fahrzeuggeschwindigkeit erfassen und einer Zentrale melden.

4.2.6 | Beschleunigung CV3

Hier lässt sich wie von einem DCC Lokdecoder her bekannt, die Beschleunigung des Fahrzeuges einstellen. Diese wird auch beim wieder auffahren in der Abstandsregelung verwendet und sollte deswegen nicht zu klein sein, sonst rast das Fahrzeug immer wieder an den Vordermann heran, bremst dann abrupt und rast wieder heran usw.. Die Werte für CV3 errechnen sich folgendermaßen: es wird alle CV3 * 8ms ein PWM Schritt beschleunigt, bis die gewünschte Geschwindigkeit erreicht ist.

4.2.7 | Bremsen CV4 und CV56

Analog zu CV3 Beschleunigung, lässt sich in CV4 die Bremsverzögerung programmieren. Es gilt wieder, CV4 * 8ms ein PWM Schritt bremsen bis die gewünschte Geschwindigkeit erreicht ist. CV56 ist für das bremsen in der Abstandsregelung zuständig. Das heißt der Wert aus CV4 wird beim normalen bremsen per Steuerinformation verwendet und der Wert aus CV56 beim bremsen über eine Abstandsinformation vom Vordermann. Da die Fahrzeuge im ungünstigen Fall schon recht dicht aufgefahren sind, gibt es die Möglichkeit hier unterschiedlich stark zu bremsen.

4.2.8 | Akkuspannung CV24 und CV25

Diese beiden CV sind für die Akkuspannung bei leerem Akku CV24 und vollem Akku CV25 zuständig. Die Werte sind die Spannungen in Volt mit einer Nachkommastelle ohne Komma.

z.B. 3,3V LiIonen Entladeschlussspannung → CV24 = 33

4,2V LiIonen Akku voll → CV25 = 42

Mit diesen Werten errechnet der Decoder die verbleibende Akkukapazität, den Korrekturwert für die Motor PWM Anpassung und die Akku leer Warnmeldung (Warnblinker ein). Die Rückmeldung der Akkukapazität erfolgt in % und kann von einem Rückmeldedecoder an eine Zentrale übertragen werden.

Zu beachten ist folgendes, die Messung der Akkuspannung ist abhängig vom Spannungsteiler R14/R15 und der internen Referenz eines jeden AVR's. Das bedeutet, dass diese Werte unter Umständen an jeden einzelnen Decoder angepasst werden müssen und so auch bei gleichem Akkutyp

hier unterschiedliche Werte in jedem Decoder nötig sein können. Empfehlung ist hier, CV24 und 25 nach der Faustregel oben einstellen und dann beim ersten auftreten der Warnblinker mal an der Ladebuchse (Außen / Mitte) die Akkuspannung zu messen. Ist diese schon unter 3,3V CV24 erhöhen bzw. ist sie noch über 3,3V CV24 absenken (das gilt natürlich nur für LiIonen Akkus). Wenn man so den optimalen Wert für CV24 gefunden hat (Warnblinker gehen bei 3,3V an) dann nimmt man die Differenz seines Wertes zu 3,3V und ändert CV25 mit der gleichen Differenz ab, das reicht von der Genauigkeit vollkommen aus. Folgend ein paar Werte für unterschiedliche Akkutypen:

Akkutyp	CV24	CV25
1x Lipo oder 1x LiIonen	33	42
1x NiCd oder 1x NiMh	9	13
2x NiCd oder 2x NiMh	18	26
3x NiCd oder 3x NiMh	27	39

4.2.9 | Decoderadresse CV1 bzw. CV17 und CV18 und CV29

Diese CV sind analog zu einem DCC Lokdecoder. Wenn in CV29 Bit5 = 0 verwendet der Decoder die kurze Adressierung nach DCC Norm über CV1 (Werte von 1 bis 127). Ist CV29 Bit5 = 1 (+32) verwendet der Decoder die lange Adressierung nach DCC Norm aus CV17 (Erweiterte Adresse Höherwertiges Byte der langen Adresse plus 192) und CV18 (Erweiterte Adresse Niederwertiges Byte der langen Adresse). Die Fahrzeugadresse wird immer als 16 Bit Wert rückgemeldet.

4.2.10 | Fahrzeugtyp CV49

Der Wert aus CV49 (0 – 255) ist frei definierbar und wird über die Fahrzeugrückmeldung übertragen, so das Fahrstrecken und Aktionen an Hand von Fahrzeugtypen ausgelöst werden können.

4.2.11 | Konfiguration aller AUX CV42 und CV70 bis CV 118

CV42

Eine Besonderheit ist der AUX_7. Dieser ist der einzige Ausgang, dessen Pegel programmierbar ist. Das heißt alle anderen AUX (auch der für den Motor) schalten bei AUX = ein den Ausgang auf GND und bei AUX = aus ist der Ausgang hochohmig. Der AUX_7 kann für externe Verwendungen auch anders schalten. Das wird in der CV42 festgelegt:

CV42	AUX eingeschaltet	AUX ausgeschaltet
0	GND	hochohmig
Bit4 = 1 (+16)	GND	+UB
Bit5 = 1 (+32)	+UB	GND

Es muss darauf geachtet werden, dass nicht Bit4 und Bit5 gleichzeitig programmiert sind, in diesem Fall setzt der Decoder intern CV42 auf 0 und startet im normalen Betriebsmode.

CV70 bis CV 118

Dies sind die CV für die Effektsteuerung der Ausgänge AUX_6 bis 10. Vorweg eine Erläuterung, wie der Decoder intern die Effekte der Ausgänge erzeugt. Für jeden Ausgang verwaltet der Decoder neun Variablen wobei die für AUX_6 bis 10 über CV programmiert werden können, die für AUX_1 bis 5 sind intern festgelegt. Die Variablen 1 bis 8 für jeden AUX stellen dabei acht Zeiten dar, wobei sich immer Ein- und Auszeit abwechseln. Variable 1 ist die erste Einzeit, Variable 2 die erste Auszeit und Variable 3 dann die zweite Einzeit usw. . Diese Zeiten werden nacheinander abgearbeitet und nach Zeit8 geht es wieder von vorn mit Zeit1 weiter. Diese Zeiten können Werte von 0 = Zeit inaktiv (Schalterpunkt wird übersprungen) und $255 * 8ms = 2,04$ Sekunden haben. Ein Beispiel, sind die Blinker welche folgendermaßen programmiert sind:

Zeit1/EIN	Zeit2/AUS	Zeit3/EIN	Zeit4/AUS	Zeit5/EIN	Zeit6/AUS	Zeit7/EIN	Zeit8/AUS
40	40	40	40	40	40	40	40

Ein Blinker funktioniert somit folgendermaßen, zuerst ist er $40 * 8ms = 0,32$ Sekunden an, dann ist er 0,32 Sekunden aus, dann wieder an ... bis Zeit8 er ist wieder 0,32 Sekunden aus. Danach beginnt es wieder von vorn mit Zeit1.

Es lassen sich damit Sequenzen aus 8 Schalterpunkten für jeden AUX programmieren, welche immer wieder abgespielt werden. Es gibt allerdings noch die neunte Variable für jeden AUX. In ihr steht, ob diese Sequenz abgespielt wird (Wert kleiner 255) oder der AUX ständig an ist (255) und somit die ersten acht Variablen ignoriert werden. Ein Wert von 254 bedeutet, dass die Sequenz immer wieder wiederholt wird, Werte von 1 – 253 sind die Anzahl der Wiederholungen der Sequenz.

Man könnte also auch folgendes machen: Zeit1 = 255 (2 Sekunden ein) Zeit2 bis 8 = 0 (Schalterpunkte werden übersprungen es wird gleich wieder Zeit1 ein ausgeführt) und Variable 9 = 3 → es soll die Sequenz 3x abgearbeitet werden → der AUX würde 3x hintereinander für 2 Sekunden zusammen also 6 Sekunden an sein und dann abschalten. Mit dieser Konstruktion, können verschiedene Effekte programmiert werden, vom Lauflicht über Blinker, Flackern, Neonstarten, Rundumlichter usw. . Für Rundumlichter stehen 2 zusammen schaltende AUX (9 und 10 bei F5) zur Verfügung. Diese werden beide mit einer Funktion geschaltet (F5), haben aber jeder einen eigenen CV Satz zur Konfiguration. Folgende Tabelle stellt die Standard Konfiguration für die AUX 6 bis 10 dar:

Zeit1/ein	Zeit2/aus	Zeit3/ein	Zeit4/aus	Zeit5/ein	Zeit6/aus	Zeit7/ein	Zeit8/aus	Wdh.
AUX_6	F6							
CV70	CV71	CV72	CV73	CV74	CV75	CV76	CV77	CV78
40	40	40	40	40	40	40	40	3
AUX_7	F7							
CV80	CV81	CV82	CV83	CV84	CV85	CV86	CV87	CV88
255	0	0	0	0	0	0	0	254
AUX_8	F8							
CV90	CV91	CV92	CV93	CV94	CV95	CV96	CV97	CV98
255	0	0	0	0	0	0	0	254
AUX_9	F5							
CV100	CV101	CV102	CV103	CV104	CV105	CV106	CV107	CV108
3	9	3	60	3	9	3	60	254
AUX_10	F5							
CV110	CV111	CV112	CV113	CV114	CV115	CV116	CV107	CV118
3	60	3	9	3	60	3	9	254

4.2.12 | Softwareupdate CV55

Der Car Decoder ist ein Open Source Eigenbauprojekt und wird als solches von hoffentlich vielen Modelleisenbahnern nachgebaut oder auch nur genutzt. Seine Funktionen werden in weiten Grenzen von den Nutzern mitbestimmt und beeinflusst, so das es sicherlich auch zukünftig neue Versionen und Erweiterungen geben wird. Abgesehen davon, bin ich „nur“ ein Hobbyprogrammierer und werde mit Sicherheit den einen oder anderen „Bug“ in die Software einbauen ;-). Aus diesem Grund, ist von vorn herein eine einfache Möglichkeit für ein Softwareupdate im OpenCarDecoder eingeplant. Den genauen Hergang findet man im Kapitel Softwareupdate (fehlt aktuell hier noch). An dieser stelle nur folgendes, steht beim Decoderstart in CV55 eine Null, startet der Decoder im normalen Betriebsmode, steht dort etwas anderes als Null, startet der Decoder im 1 – wire Softwareupdatemode. In diesem bleibt er auch, bis CV55 wieder auf Null gesetzt ist! Wenn man also die Decodersoftware Updaten will, schreibt man in CV55 eine 1 und starte den Decoder neu (aus / ein). Baut jetzt eine 1- wire Verbindung auf, wechselt die Software und das EEPROM File !! (damit CV55 wieder auf Null gesetzt ist) und startet dann den Decoder wieder im Normalmode neu.

Anhang