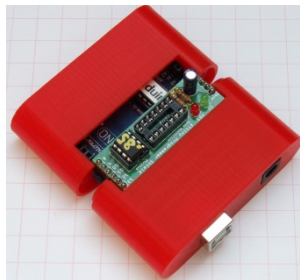


Platinen-Dokumentation



Platinen zum Einsatz in und an der Modellbahn von Bodo Noethlich
Arduino, ATtiny & Co



Inhaltsverzeichnis

VORBEMERKUNG / ALLGEMEINE HINWEISE	4
ÜBERSICHT PLATINENMERKMALE	5
Platinen für Arduino Nano	5
Sonstige Platinen	5
ARDUINO NANO	6
Nano-Dekoder-3 (rev.1, 29.12.2015)	6
Fritzing-Zeichnung	6
Platinen-Layout	6
Freigabe-Link	6
Benötigte Bauteile	6
Hinweise zur Bestückung	6
Hinweise zum Anschluss und zum Betrieb	6
Nano-Dekoder-4 (rev.3, 26.05.2018)	7
Fritzing-Zeichnung	7
Platinen-Layout	7
Freigabe-Link	7
Benötigte Bauteile	7
Hinweise zur Bestückung	7
Hinweise zum Anschluss und zum Betrieb	8
Nano-Dekoder-5 (rev.2, 26.05.2018)	8
Fritzing-Zeichnung	8
Platinen-Layout	8
Freigabe-Link	9
Benötigte Bauteile	9
Hinweise zur Bestückung	9
Hinweise zum Anschluss und zum Betrieb (aktualisiert 12.08.2018)	9
Nano-Dekoder-5 (rev.3, Planungsliste, aktualisiert 12.08.2018)	13
Nano-Dekoder-6 (rev.1, 06.01.2019)	13
Fritzing-Zeichnung	14
Platinen-Layout	14
Freigabe-Link	14
ATTINY MIKROPROZESSOREN 13/44/45/84/85	15
ATtiny Programmer Shield (rev.1, 11.04.2018)	15
Fritzing-Zeichnung	15
Platinen-Layout	15
Freigabe-Link	15
Benötigte Bauteile	15
Hinweise zur Bestückung	15
Hinweise zu Anschluss und Betrieb	16
ATtiny-Dekoder-3 (rev.1, 24.10.2018)	16
Fritzing-Zeichnung	17
Platinen-Layout	17
Freigabe-Link	17
Hinweise zur Bestückung	17
Hinweise zu Anschluss und Betrieb	17
ATtiny-Dekoder-5	17



KLEINE HELFERLEIN	18
6n137-Adapter (rev.1, 29.12.2015)	18
Fritzing-Zeichnung	18
Platinen-Layout	18
Freigabe-Link	18
Benötigte Bauteile	18
Hinweise zur Bestückung	18
Hinweise zu Anschluss und Betrieb	18
Schieberegister zweifach (rev.1, 25.11.2018)	19
Fritzing-Zeichnung	19
Platinen-Layout	19
Freigabe-Link	19
Benötigte Bauteile	19
Hinweise zur Bestückung	19
Hinweise zu Anschluss und Betrieb	20
Schieberegister zweifach (rev.2, 06.12.2018)	20
Freigabe-Link	20
Servo-Filter (rev.5, 18.06.2018)	21
Fritzing-Zeichnung	21
Platinen-Layout	21
Freigabe-Link	21
Benötigte Bauteile	21
Hinweise zur Bestückung	21
Hinweise zu Anschluss und zum Betrieb	21
Bremsmodul Typ „BM1“ für Lenz ABC (rev.2, 23.05.2020)	23
Fritzing-Zeichnung	23
Platinen-Layout	23
Freigabe-Link	23
Benötigte Bauteile	23
Hinweise zur Bestückung	23
Hinweise zu Anschluss und zum Betrieb	23



Vorbemerkung / Allgemeine Hinweise

Im Rahmen meiner Hobby-Basteleien entstanden in den letzten Jahren einige Platinen zum Einsatz in und an der Modellbahn. Viele davon als Ergänzung zu den Mikroprozessoren des Typs ATtiny und rund um die Arduino-Familie.

In erster Linie zur Dokumentation für mich selbst, aber auch für Modellbahn-Kollegen, die meine Platinen ebenfalls einsetzen möchten, wird dieses Dokument als Hilfe zu Bestückung und Anschluss geführt. Da die Platinen nach Bedarf weiterentwickelt werden, ist für jedes Kapitel ein eigenes Aktualisierungsdatum angegeben.

Sie sollten dieses Dokument direkt von <http://moba.noethlich.info> oder über einen Direktlink auf <https://www.noethlich.net/downloads/platinen-dokumentation.pdf> in einem der einschlägigen Foren (z.B. stummiforum.de) heruntergeladen haben – andernfalls handelt es sich wahrscheinlich um eine veraltete Version und eventuell sogar um eine nicht genehmigte Kopie. Laden Sie dann bitte die aktuelle Version über den o.a. Link.

Alle Platinen wurden nach bestem Wissen erstellt und vor Freigabe getestet. Im Digital-Bereich kommt dazu in der Regel eine Mobile Station 2 der Firma Märklin zum Einsatz, ansonsten Stecker-Netzteile 5-12 V. Trotzdem sind Fehler natürlich nie ganz auszuschließen. Die Verwendung erfolgt daher immer auf eigene Gefahr.

Über Korrekturhinweise an bodo.noethlich@gmx.de freue ich mich ggf. natürlich auch ...

Bei den Bauteile-Listen beziehe ich mich fast ausschließlich auf das Sortiment der Firma Reichelt-Elektronik. Dies stellt keine explizite Empfehlung dar, sondern ist für mich einfach am bequemsten. Natürlich gibt es sehr viele andere Bezugsquellen. Bei den Preisen ist zu berücksichtigen, dass es gerade bei Kleinteilen Mengenrabatte gibt, die die Bestellung z.B. einzelner Widerstände völlig unsinnig machen (100 Stück kosten z.B. nicht wesentlich mehr als 10 Stück).



Übersicht Platinenmerkmale

Zur schnellen Orientierung ein paar Rahmen-Kriterien ... die Freigabelinks werden bei der PDF-Erstellung anscheinend „verstümmelt“ und müssen dann entweder manuell wieder zusammengesetzt oder aus den betreffenden detaillierten Beschreibungen (dort sind sie auch im PDF vollständig) aufgerufen werden.

Platinen für Arduino Nano

	Nano-Dekoder-3 https://aisler.net/pl/LQZRNLQG	Nano-Dekoder-4	Nano-Dekoder-5 https://aisler.net/pl/HAJDKFU
Abmessungen in mm	49 x 18	47 x 18	50 x 43
SMD-Bauteile notwendig	nein	ja	nein
Separate Versorgungsspannung nötig	7V-12V	7V-12V	nein
Verstärkte Ausgänge	nein	8 (externe Spannung)	8 (Digital-Spannung)

Sonstige Platinen

	ATtiny-Programmer https://aisler.net/pl/MIACIAJU	6N137-Adapter	Servo-Filter https://aisler.net/pl/MKDFUCES
Abmessungen in mm	51 x 17	22 x 11	18 x 8
SMD-Bauteile notwendig	nein	ja	ja
Separate Versorgungsspannung nötig	nein	nein	nein



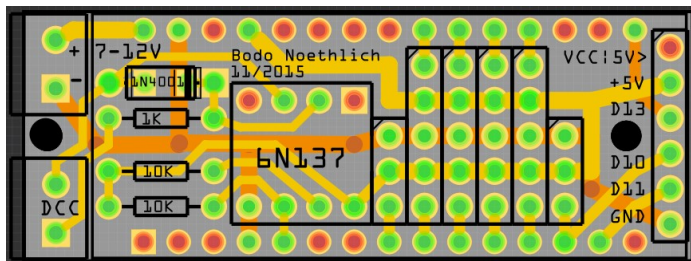
Arduino Nano

Nano-Dekoder-3 (rev.1, 29.12.2015)

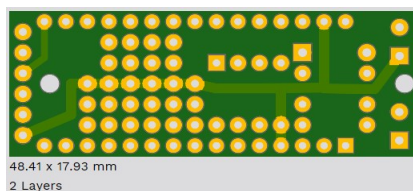
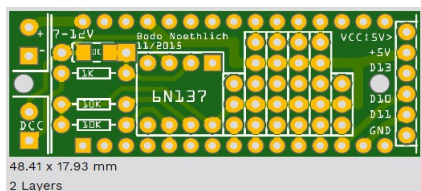
Ziel bei dieser Platine war es, den Optokoppler zum Einspeisen des DCC-Signals möglichst platzsparend zwischen den „Beinen“ des Arduino-Nano unterzubringen und die Ports möglichst flexibel nutzen zu können.

Die Bestückung erfolgt komplett mit konventionellen Bauteilen (keine SMD-Komponenten).

Fritzing-Zeichnung



Platinen-Layout



Freigabe-Link

Unter <https://aisler.net/p/LQZRNLQG> kann diese Platine bei AISLER bestellt werden.

Benötigte Bauteile

	z.B. Artikel-Nr. bei Reichelt Elektronik	
Optokoppler 6N137	6N 137	0,52 €
2 x Widerstand 10K 1/4W	1/4W 10K	0,10 €
1 x Widerstand 1K oder 1,5K 1/4W	1/4W 1,0K oder 1/4W 1,5K	0,10 €
1 x Diode 4148, alternativ Leuchtdiode SMD 1206 oder 0804		

Optional: Stiftleiste / Buchsenleiste RM 2,54mm, Schraubklemmen RM 3,5mm, Servo-Anschlusskabel

Hinweise zur Bestückung

Die Diode ist auf dem (bezogen auf die Abbildung) obersten Feld mit Stromflußrichtung „nach rechts“ zu montieren. Der Optokoppler ist mit der Markierung „rechts oben“ (bezogen auf die Abbildung) zu montieren. Statt der Standard-Diode kann eine SMD-LED verwendet werden, die dann im Betrieb das Anliegen des DCC-Signals anzeigt.

Bei Verzicht auf IC-Sockel und Buchsenleisten für den Arduino Nano ist die gesamte Baugruppe klein genug, um in einem „Wattenscheider Schacht“ (FREMO) Platz zu finden.

Hinweise zum Anschluss und zum Betrieb

Die NMRA-DCC-Bibliothek unterstützt den Empfang des DCC-Signals über einen Pin des Arduino und die Rückmeldung z.B. zur CV-Programmierung über einen zweiten Pin. Dieser Dekoder enthält nur den Optokoppler zum Empfang des DCC-Signals, da ich meine Software i.d.R. direkt auf die Anwendung abstimme.



Was für kommerzielle Dekoder natürlich nicht mehr zeitgemäß ist, spart mir Hardware und den entsprechenden Port.

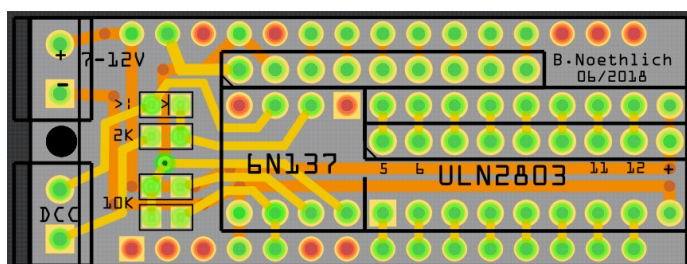
Der 6-polige Anschluss auf der rechten Seite ist zur Kombination mit Schieberegister-Platinen eines Modellbahnkollegen vorgesehen.

Nano-Dekoder-4 (rev.3, 26.05.2018)

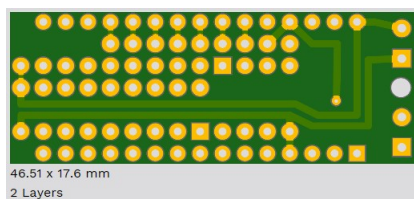
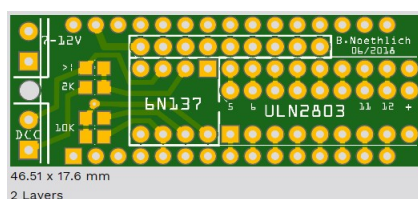
Als Weiterentwicklung des Nano-Dekoder-3 wurde zusätzlich noch ein Transistor-Array auf der Platine untergebracht.

Wegen der zusätzlichen Leiterbahnen musste bei der Beschaltung des Optokopplers nun aber auf SMD-Bauteile ausgewichen werden.

Fritzing-Zeichnung



Platinen-Layout



Freigabe-Link

Unter <https://aisler.net/p/BMPDYRJX> kann diese Platine bei AISLER bestellt werden.

Benötigte Bauteile

	z.B. Artikel-Nr. bei Reichelt Elektronik	
Optokoppler 6N137	6N 137	0,52 €
Darlington-Transistor-Array ULN2803	ULN 2803A	0,29 €
2 x Widerstand 10K, SMD 0805	RND 0805 1 10K	0,02 €
1 x Widerstand 1K oder 1,5K, SMD 0805	RND 0805 1 1,5K	0,02 €
1 x Leuchtdiode SMD 0805		

Optional: Stiftleiste / Buchsenleiste RM 2,54mm, IC-Sockel 8 & 18, Servo-Anschlusskabel

Hinweise zur Bestückung

Die Diode ist auf dem (bezogen auf die Abbildung) obersten Feld mit Stromflußrichtung „nach rechts“ zu montieren. Der Optokoppler ist mit der Markierung „rechts oben“ (bezogen auf die Abbildung) zu montieren.

Soll zum Anschluss an die Ausgänge eine Stiftleiste verwendet werden, so ist diese vor der Montage des ULN2803 oder dessen Sockel einzulöten – später sind die entsprechenden Lötstellen verdeckt.

Bei Verzicht auf IC-Sockel und Buchsenleisten für den Arduino Nano ist die gesamte Baugruppe klein genug, um in einem „Wattenscheider Schacht“ (FREMO) Platz zu finden.



Hinweise zum Anschluss und zum Betrieb

Die NMRA-DCC-Bibliothek unterstützt den Empfang des DCC-Signals über einen Pin des Arduino und die Rückmeldung z.B. zur CV-Programmierung über einen zweiten Pin. Dieser Dekoder enthält nur den Optokoppler zum Empfang des DCC-Signals, da ich meine Software i.d.R. direkt auf die Anwendung abstimme.

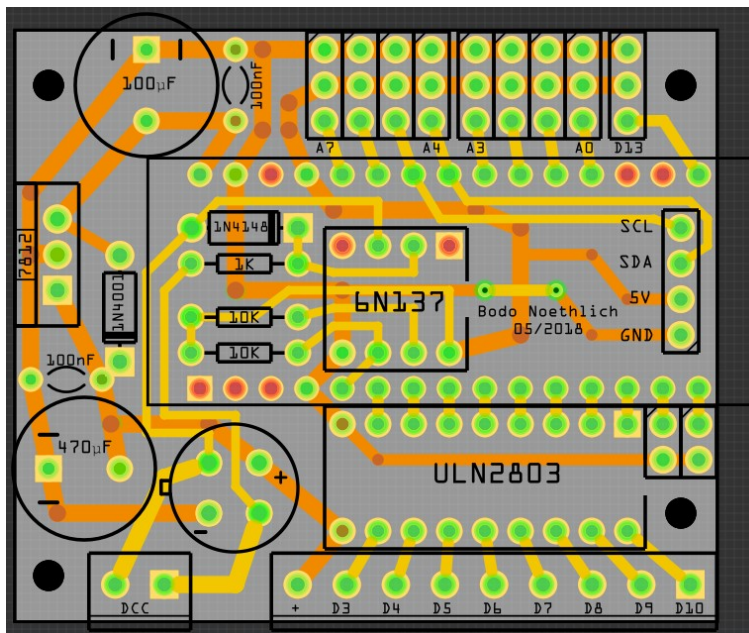
Was für kommerzielle Dekoder natürlich nicht mehr zeitgemäß ist, spart mir Hardware und den entsprechenden Port.

Nano-Dekoder-5 (rev.2, 26.05.2018)

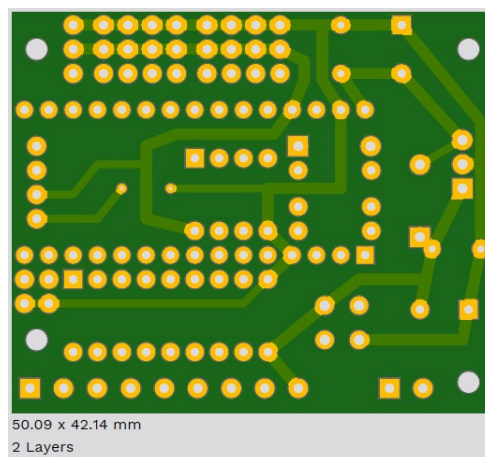
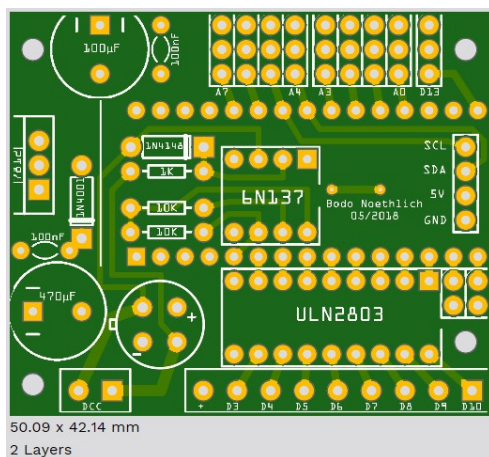
Für den direkten Anschluss an die Digitalspannung ist die Variante -5 entworfen. Zur Versorgung des Arduino Nano ist daher ein Spannungsregler 9-12V auf der Platine integriert. Die verstärkten Ausgänge werden direkt mit der gleichgerichteten Digitalspannung beschaltet und sind damit insbesondere auch für alte Glühlämpchen-Signale geeignet.

Die Bestückung erfolgt komplett mit konventionellen Bauteilen (keine SMD-Komponenten).

Fritzing-Zeichnung



Platinen-Layout





Freigabe-Link

Unter <https://aisler.net/p/HAJDCKFU> kann diese Platine bei AISLER bestellt werden.

Benötigte Bauteile

	z.B. Artikel-Nr. bei Reichelt Elektronik	
Optokoppler 6N137	6N 137	0,52 €
Darlington-Transistor-Array ULN2803	ULN 2803A	0,29 €
Gleichrichter 2A	B80C2000RUND	0,21 €
Spannungsregler 9V oder 12V	µA 7812	0,23 €
2 x Widerstand 10K 1/4W	1/4W 10K	0,10 €
1 x Widerstand 1K oder 1,5K 1/4W	1/4W 1,0K oder 1/4W 1,5K	0,10 €
2 x Diode (4148 und/oder 4001)		
1 x Kondensator 470µF	RAD 470/35	0,11 €
1 x Kondensator 100µF	RAD 100/35	0,05 €
2 x Kondensator 100nF	KERKO 100N	0,04 €
Optional: Stiftheiste / Buchsenleiste RM 2,54mm, Schraubklemmen RM 3,5mm, IC-Sockel 8 & 18		

Hinweise zur Bestückung

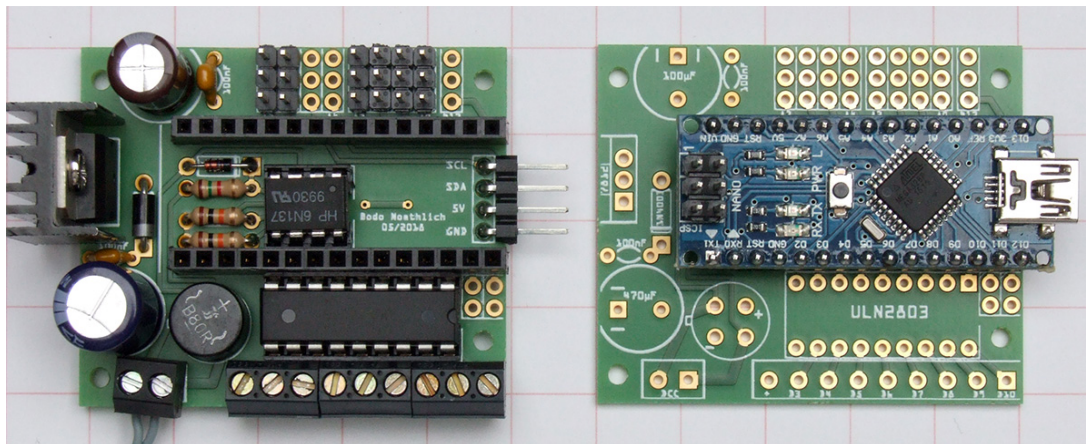
Die Platine ist inzwischen sehr gut beschriftet, so dass kaum Fragen offen bleiben sollten.

Die Löcher für den Gleichrichter stehen etwas zu eng, so dass dieser sich nur „schwebend“ einlöten lässt. Er bleibt damit aber trotzdem auf Höhe des gesockelten ULN2803 und niedriger als der Kondensator und der Nano selbst.

Anstelle des 7812 Spannungsreglers kann auch ein Pin-kompatibler Step-Down-Regler eingesetzt werden – diese Bauteile gibt es aus Fern-Ost bereits relativ günstig z.B. über eBay.

Bei der Bestückung mit Schraubklemmen ist es empfehlenswert, mit diesen zu beginnen, dann bündig den Sockel für das ULN2803 einzulöten und wieder daran bündig die untere Buchsenleiste für den Arduino Nano. Dann ggf. den Sockel für den Optokoppler, den I2C-Anschluss und die 3 Widerstände und die Diode unter dem Nano, Zuletzt dann die zweite Buchsenleiste.

Der Optokoppler ist mit der Markierung „rechts oben“ (bezogen auf die Abbildung) zu montieren.



Hinweise zum Anschluss und zum Betrieb (aktualisiert 12.08.2018)

Die NMRA-DCC-Bibliothek unterstützt den Empfang des DCC-Signals über einen Pin des Arduino und die Rückmeldung z.B. zur CV-Programmierung über einen zweiten Pin. Dieser Dekoder enthält nur



den Optokoppler zum Empfang des DCC-Signals, da ich meine Software i.d.R. direkt auf die Anwendung abstimme.

Was für kommerzielle Dekoder natürlich nicht mehr zeitgemäß ist, spart mir Hardware und den entsprechenden Port. Da der für den Rückkanal häufig genutzte Port D4 bei dieser Platine mit zu den genutzten Ausgängen gehört, muss der ACK-Pin bei „Fremdsoftware“ ggf. auf einen anderen (freien) Port gelegt werden.

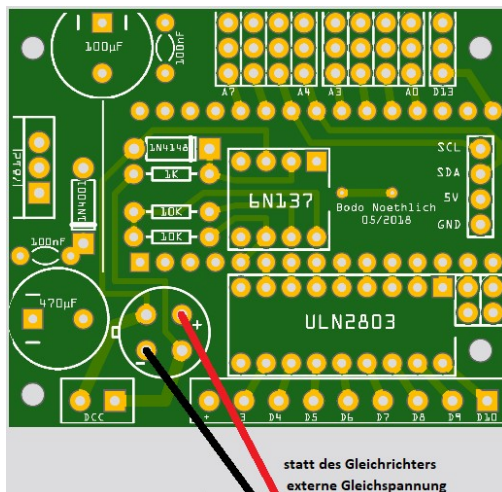
Wenn der I2C-Anschluss (kompatibel zu den üblichen Expansion-, Breakout- oder Sensor-Shield genannten Platinen – 4 Pins SCL/SDA/5V/Masse) genutzt werden soll, dürfen A4 und A5 natürlich nicht doppelt belegt werden.

A6 und A7 sind beim Arduino Nano reine Analog-Eingänge und daher auch nur als solche zu verwenden – für die Spannungsteiler für die Programmiermodi bieten sie sich daher ebenso an, wie für zusätzliche Sensoren (z.B. Helligkeit oder Temperatur).

Externe Spannungsversorgung

Der Dekoder ist dafür ausgelegt, ohne externe Spannungsversorgung auszukommen. Trotzdem mag es Gründe geben, dieses Layout statt der kleineren Platinen (Nano-Dekoder-3 und -4) zu verwenden, und trotzdem mit externer Spannungsversorgung zu arbeiten.

Zu berücksichtigen ist, dass diese Gleichspannung dann auch die ist, die am ULN2803 anliegt. Dann gibt es zwei Möglichkeiten: Bei Spannungen über 12V entfällt nur der Gleichrichter. Dort, wo dessen Ausgänge eingelötet werden, wird direkt die externe Gleichspannung angelegt.



Liegt die externe Spannung im Bereich von 7-12V, kann auch auf den Spannungsregler und dessen Komponenten verzichtet werden. Dazu sind Ein- und Ausgang des Spannungsreglers zu überbrücken und die beiden 100nF-Kondensatoren, die Schutzdiode und einer oder sogar beide Elkos können entfallen. Die Platine ist dann aber schon (bei mehr Fläche und höherem Preis) auf das technische Niveau von Nano-Dekoder-3 oder -4 reduziert ...

Beispiel „Arduinobasierter DIY Zubehörcodecorder“ (aktualisiert 12.08.2018)

Als Software kann zum Beispiel der „Arduinobasierte DIY Zubehörcodecorder“ aus dem Stummforum (<https://www.stummiforum.de/viewtopic.php?f=5&t=144916>) installiert werden. Ich nutze hier folgende Werte in der Konfigurationsdatei `DCC_Zubehoerdecoder.h`:

```
const byte dccPin    = 2;
const byte ackPin    = NC;    (geht bei Version 6.0 nicht mehr ?)
#define FIXMODE INIMODE
const byte betrModeP = A7;
const byte resModeP  = A6;
// #define ENCODER_AKTIV
```



Als Beispielkonfiguration hier zunächst ein Ausfahrtsignal auf Adresse 5/6 (Port 3-7, Relais an Port A3) mit Vorsignal am Mast (Adresse 7 und 22, zweibegriffig an Port 9&10), einem Servo-Antrieb an Adresse 8 und Andreaskreuzen auf Adresse 9 (LEDs direkt an Port A1/A2)

```
// Ausfahrtsignal HP00,HP1,SH1,HP2 mit Vorsig.am Mast (Viessm.H0 4016)
const byte iniTyp[] = {FSIGNAL2,FSIGNAL0, FVORSIG, FSERVO, FSTATIC};
const byte out1Pins[] = { /*rt*/ 3, /*rt*/ 4, /*ge*/ 9, A0, A1};
const byte out2Pins[] = { /*gn*/ 5, /*ws*/ 7, /*gn*/ 10, NC, A2};
const byte out3Pins[] = { /*ge*/ 6, A3, NC, NC, NC};
const byte iniFmode[] = { 0, 0b000100, 0, SAUTOOFF, 0b000111};
const byte iniPar1[] = { 0b001001, 0b110001, 0b000001, 30, 100};
const byte iniPar2[] = { 0b100010, 0b100110, 0b000010, 150, 100};
const byte iniPar3[] = { 3, 0, 22, 8, 50};
const byte iniPar4[] = { 0b000101, 0, 0, 0, 0};
```

Möchte man die Software ohne festen „INIMODE“ nutzen, so kann man sich zwei Steckbrücken für betrModeP basteln, die den Pullup-Widerstand und die passenden Spannungsteiler-Widerstände enthalten. Möchte man für Servos den Drehencoder nutzen, so benötigt man auch für resModeP einen Pullup-Widerstand (10K) und optional einen Taster gegen Masse.

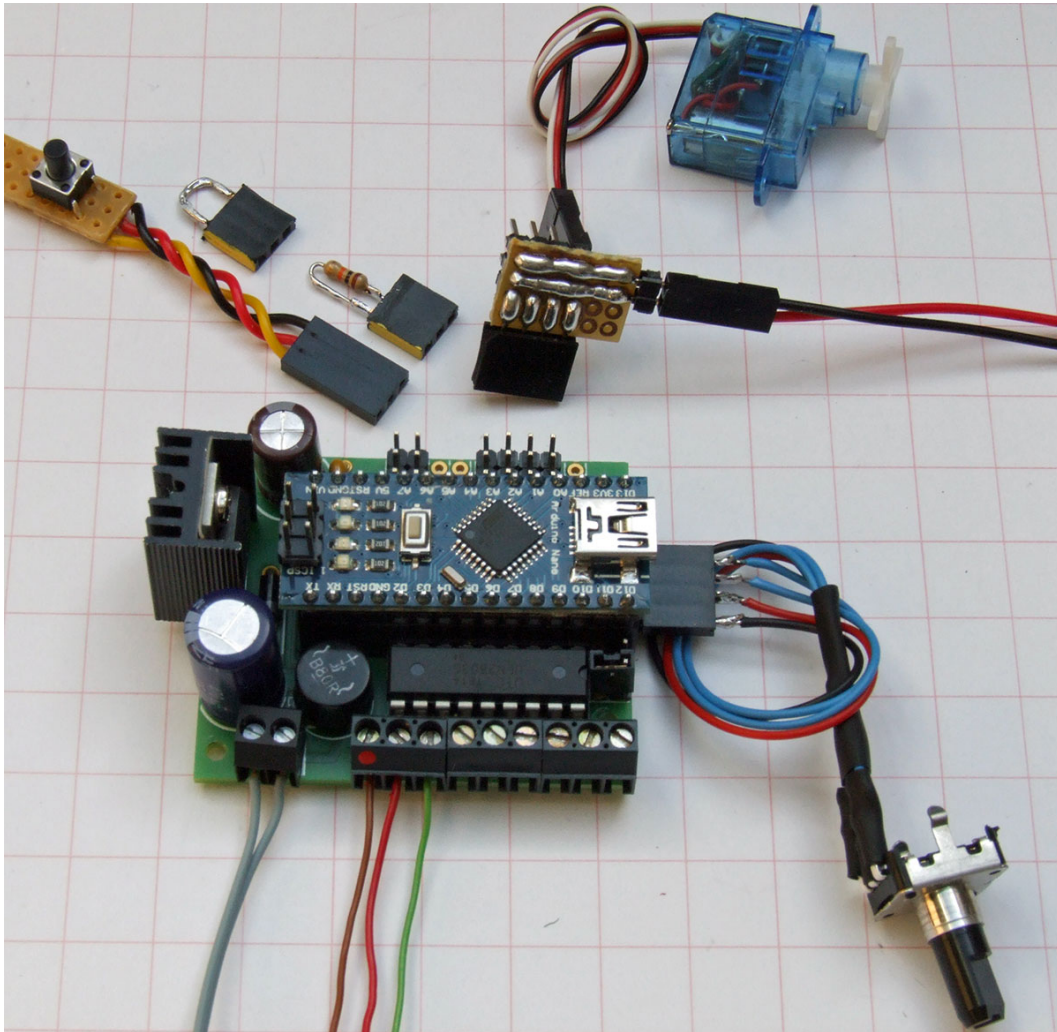
Als Pullup-Widerstand kann man gegebenenfalls auch auf der Rückseite einen 10K SMD-Widerstand zwischen die Pins (Port und 5V) löten – darf man nur bei der Beschaltung später nicht vergessen, da ein zweiter paralleler Widerstand dann einen kleineren Wert als vorgesehen ergibt ...

Dann kann man mit der Konfiguration in `DCC_Zubehoerdecoder.h` wie folgt arbeiten:

```
const byte dccPin      = 2;
const byte ackPin      = NC;      (geht bei Version 6.0 nicht mehr ?)
const byte betrModeP    = A7;
const byte resModeP     = A6;
#define ENCODER_AKTIV
const byte encode1P     = A5;
const byte encode2P     = A4;
```

Als Encoder habe ich erfolgreich den kleinen Drehimpulsgeber von Alps, den Reichelt unter STEC12E07 vertreibt, getestet.

Die folgende Abbildung zeigt die „kleinen Helferlein“, die ich mir zum Betrieb der Software des Zubehör-Dekoders auf meiner Platine gebastelt habe. Neben den Jumpers für A6 und A7 und dem Drehimpulsgeber an A4 und A5 ist dabei auch noch eine Adapterplatine zur externen Spannungsversorgung der Servos an A0 bis A3 zu sehen (Masse der externen Spannungsversorgung muss mit Masse des Arduino verbunden sein):

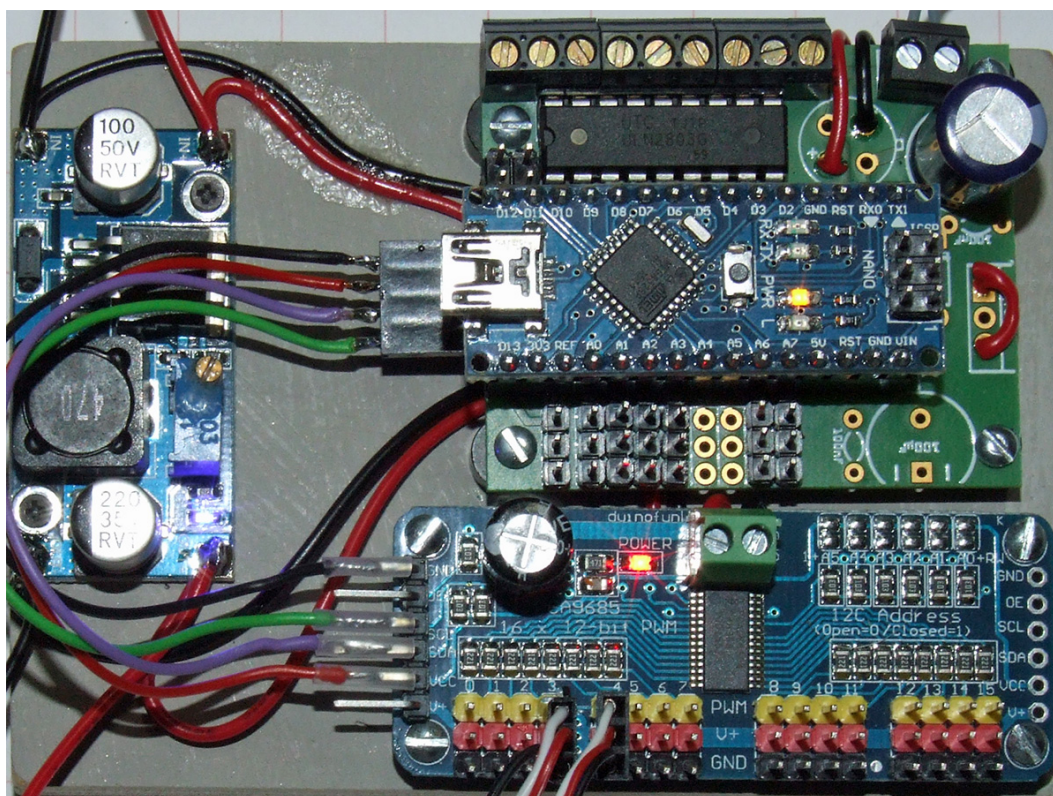


Beispiel mit Adafruit 16-Kanal-Servo-Treiber PCA9685 (aktualisiert 12.08.2018)

Der 16-Kanal-Servo-Treiber (bei Reichelt unter DEBO MOTODRIVER4, als Fern-Ost-Nachbau deutlich billiger) wird per I2C angesteuert und kann direkt 16 Servos steuern – bei Bedarf kann man auch noch weitere Bausteine an den I2C-Bus anschließen.

Fertige Dekoder-Software gibt es dafür zwar m.W. noch nicht, eine rudimentäre Software zum Umschalten von 16 Servos über 16 Adressen habe ich jedoch bereits lauffähig. Eine interessante Anwendung wären z.B. Schuppentore (zwei Flügel und Relais-Ausgang pro Adresse) oder Funktionsmodelle, die mehr als 4 Servos auf engem Raum verwenden.

Die Abbildung zeigt außerdem noch mal die externe Spannungsversorgung der Platine. „Links oben“ kommen 12V Gleichspannung rein, die dann auf die Platine und den Step-Down-Regler (links im Bild) verteilt werden. Der Regler ist auf 5V Ausgangsspannung eingestellt und versorgt damit die Treiber-Platine (Schraubklemme). Die etwas unübersichtlich enge Montage ist hier der Unterbringung im Raaco-Sortimentskasten geschuldet ...



Zunächst nur Sammlung der Vorschläge ...

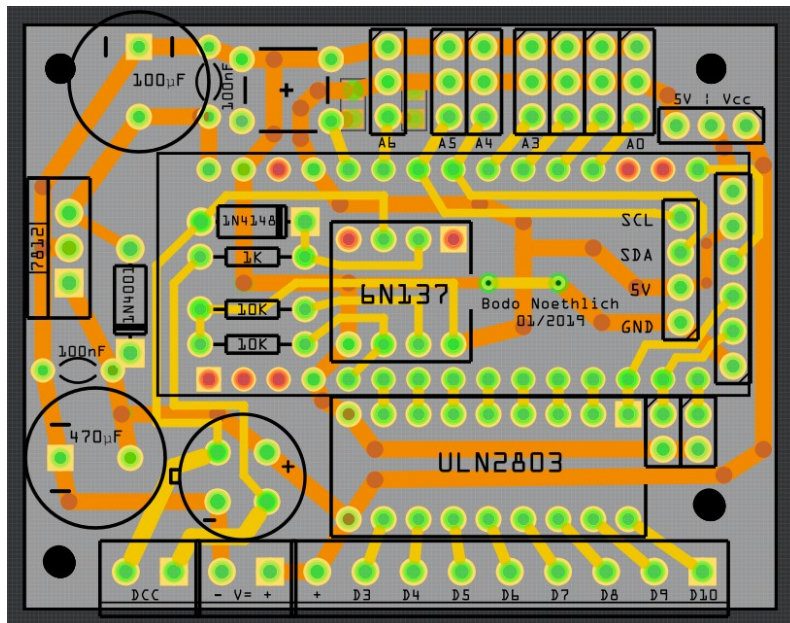
- Nano-Dekoder-6 (rev.1, 06.01.2019)**

Zusätzlich realisiert wurde ein Anschluss für die Schieberegister-Platinen, Pull-Up-Option für A6 und A7, Taster an A7, Schraubklemme für externe Gleichspannung (der Gleichrichter darf in diesem Fall nicht montiert werden). Er ist dadurch noch etwas besser an den „Stummi-DIY-Zubehör-Dekoder“ (<https://www.stummiforum.de/viewtopic.php?f=5&t=144916>) angepasst, aber direkt auch wieder etwas größer geworden.

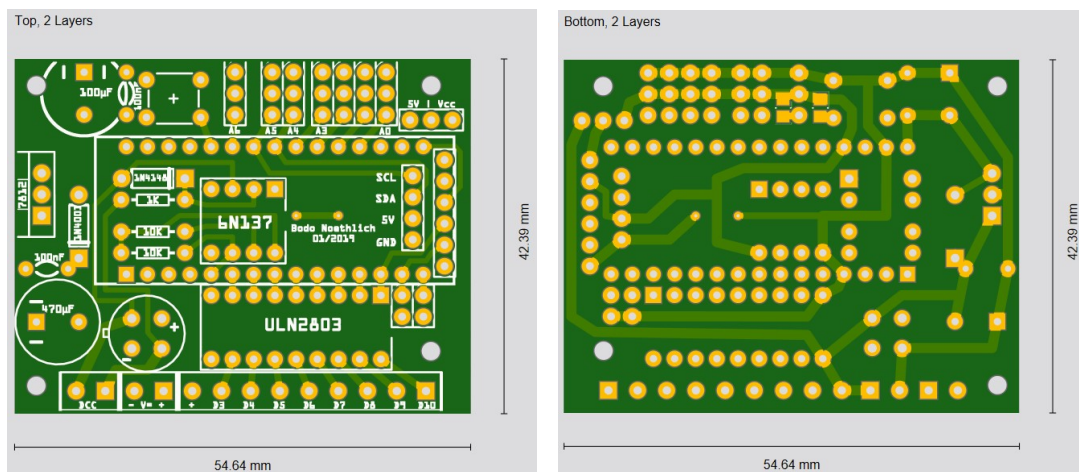
Die Bestückung erfolgt bis auf die Pull-Up-Widerstände für A6 und A7 (als SMD 0805) komplett mit konventionellen Bauteilen.



Fritzing-Zeichnung



Platinen-Layout



Freigabe-Link

Die ersten Test-Platinen sind in der Produktion – Freigabe erfolgt nach positivem Test der Platine.

Unter <https://aisler.net/p/UAJDMRJD> kann diese Platine bei AISLER bestellt werden.



ATtiny Mikroprozessoren 13/44/45/84/85

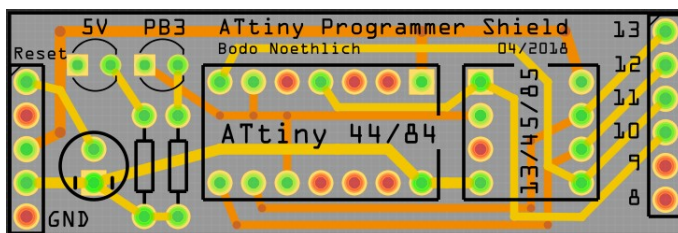
ATtiny Programmer Shield (rev.1, 11.04.2018)

Der Arduino kann nicht nur selbst als aktives Bauteil eingesetzt werden, sondern mit Hilfe eines in den Beispielen mitgelieferten Sketches auch als ISP-Programmer für die kleinen ATtiny-Mikroprozessoren (13/44/45/84/85) dienen.

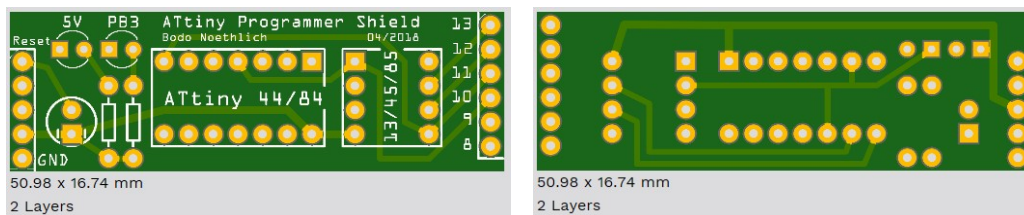
Um den Vorgang einfach und bequem zu gestalten, habe ich ein entsprechendes Programmer-Shield entworfen, welches einfach auf den Arduino UNO aufgesteckt werden kann.

Während der 10µF-Kondensator und der IC-Sockel für den Mikroprozessor für die Funktion erforderlich sind, können die LEDs mit ihren Vorwiderständen auch weggelassen werden – die Funktion des Programmers wird dadurch nicht beeinträchtigt.

Fritzing-Zeichnung



Platinen-Layout



Freigabe-Link

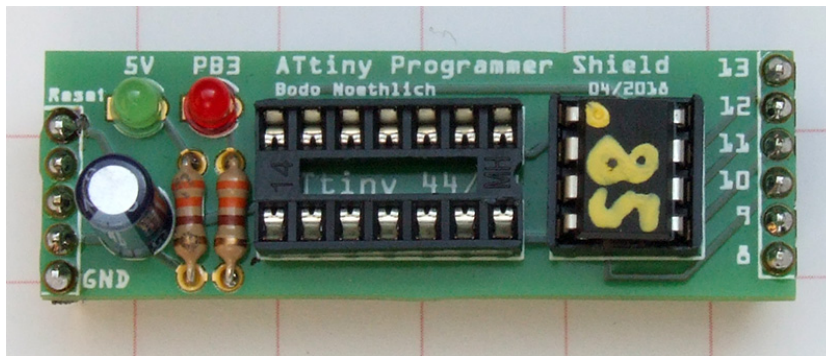
Unter <https://aisler.net/p/MIACAJU> kann diese Platine bei AISLER bestellt werden.

Benötigte Bauteile

	z.B. Artikel-Nr. bei Reichelt Elektronik	
1 x Kondensator 10µF	RAD 10/63	0,02 €
Stecksocket für ATtiny (8- und /oder 14-polig)		
Stiftleiste RM 2,54mm (einmal 5- und einmal 6-polig)		
Optional: 2 x LED 3mm und 2 x Vorwiderstand 330 oder 470 Ohm		

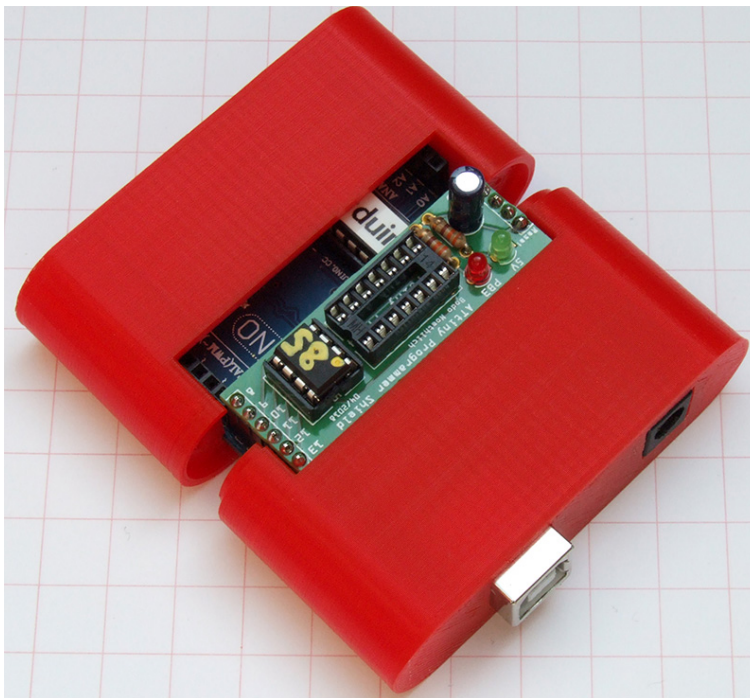
Hinweise zur Bestückung

Für die Bestückung sollten zusätzlich zu den Abbildungen keine weiteren Hinweise erforderlich sein:



Hinweise zu Anschluss und Betrieb

Die Platine wird einfach auf den als ISP-Programmer vorgesehenen Arduino aufgesteckt.



Das passende Gehäuse schützt vor Kurzschlüssen auf dem Bastel-Tisch – ein Holzbrettchen als Unterlage täte es auch.

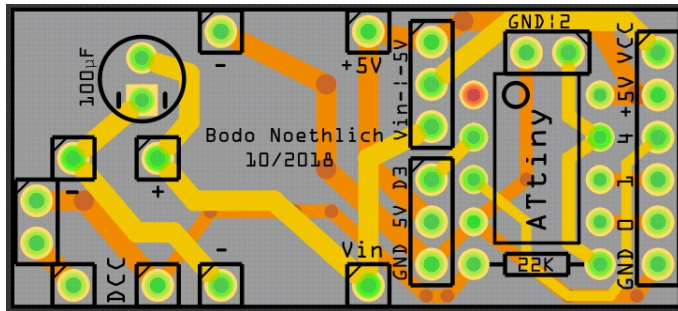
ATtiny-Dekoder-3 (rev.1, 24.10.2018)

Diese Platine orientiert sich am „DigiSpark“. Die Versorgungsspannung wird über einen handelsüblichen kleinen StepDown-Regler realisiert, auf Port 2 wird über 22 KOhm das Digital-Signal zur Auswertung durch die NmraDcc-Lib eingespeist und die Ausgänge sind kompatibel zur Schieberegister-Platine herausgeführt. Port 3 kann zusammen mit 5V und Masse als Analog-Eingang oder als Ausgang genutzt werden.

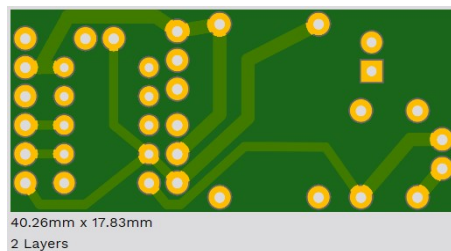
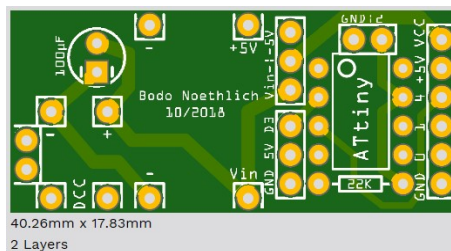
Die Bestückung erfolgt komplett mit konventionellen Bauteilen (keine SMD-Komponenten).



Fritzing-Zeichnung



Platinen-Layout

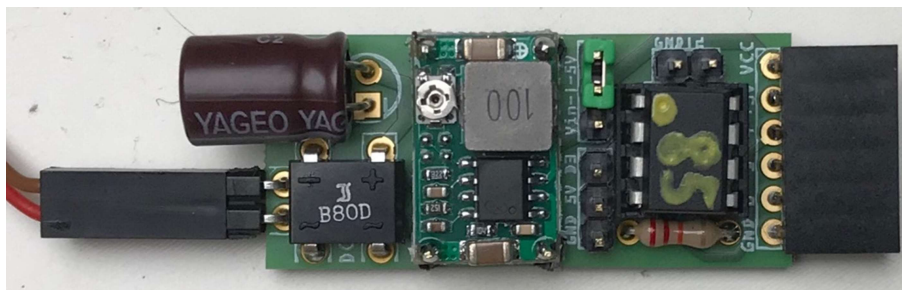


Freigabe-Link

Unter <https://aisler.net/p/MOENVNAL> kann diese Platine bei AISLER bestellt werden.

Hinweise zur Bestückung

Für die Bestückung sollten zusätzlich zu den Layout-Zeichnungen und der Abbildung keine weiteren Hinweise erforderlich sein:



Hinweise zu Anschluss und Betrieb

Achtung: den StepDown-Regler unbedingt vor Einstecken des ATtiny auf 5V einstellen und fixieren – im Auslieferungszustand ist i.d.R. eine deutlich höhere Spannung eingestellt.

Eine kleine Demonstration der Dekoders im Zusammenspiel mit den Schieberegistern gibt es z.B. bei Youtube unter <https://www.youtube.com/watch?v=u9EztBKtRLM>.

ATtiny-Dekoder-5



Kleine Helferlein

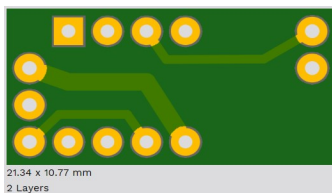
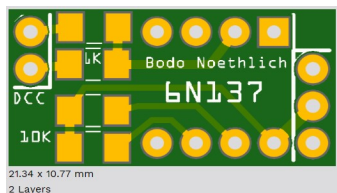
6n137-Adapter (rev.1, 29.12.2015)

Sozusagen als „Abfallprodukt“ der Platinen für den Arduino Nano entstand diese Adapterplatine zum Anschluss des Optokopplers an die üblichen Expansion-, Breakout- oder Sensor-Shield genannten Platinen

Fritzing-Zeichnung



Platinen-Layout



Freigabe-Link

Unter <https://aisler.net/p/HTSVZFVA> kann diese Platine bei AISLER bestellt werden.

Benötigte Bauteile

z.B. Artikel-Nr. bei Reichelt Elektronik

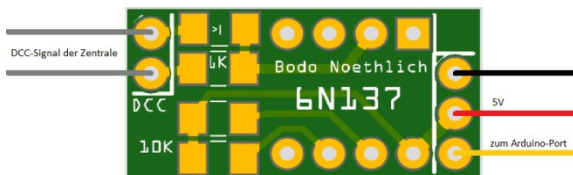
Optokoppler 6N137	6N 137	0,52 €
2 x Widerstand 10K, SMD 1206 oder 0805	RND 0805 1 10K	0,02 €
1 x Widerstand 1K oder 1,5K, SMD 1206 oder 0805	RND 0805 1 1K	0,02 €
1 x Leuchtdiode SMD 1206 oder 0804, alternativ: 1 x Diode 4148 SMD		
Optional: Stiftheiste / Buchsenleiste RM 2,54mm, Servo-Anschlusskabel		

Hinweise zur Bestückung

Die Diode ist auf dem (bezogen auf die Abbildung) obersten Feld mit Stromflußrichtung „nach rechts“ zu montieren. Der Optokoppler ist mit der Markierung „rechts oben“ (bezogen auf die Abbildung) zu montieren.

Hinweise zu Anschluss und Betrieb

Über den Anschluss auf der linken Seite wird das DCC-Signal eingespeist, der Anschluss auf der rechten Seite wird mit dem Arduino verbunden. Dabei gilt „von oben nach unten“ Masse / 5V / Signal.





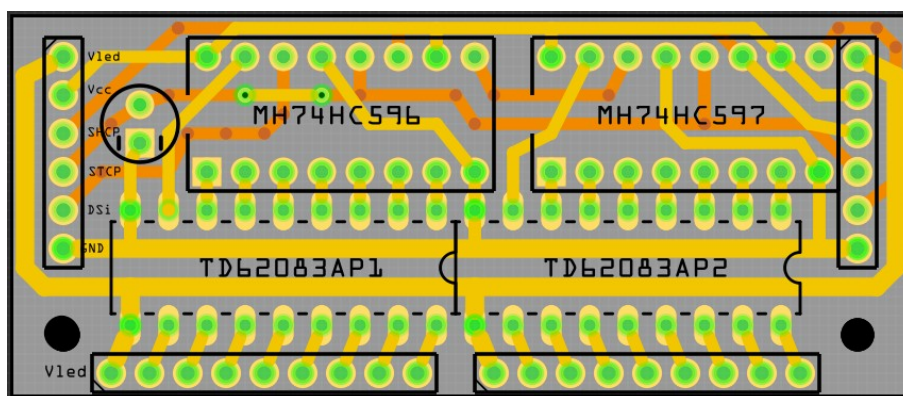
Schieberegister zweifach (rev.1, 25.11.2018)

Das Kaskadieren von Schieberegistern stellt eine einfache Methode der Porterweiterung für Mikrocontroller dar, zumal auch die Ansteuerung über Bit-Operationen auf die Ausgabewerte sehr elegant und platzsparend hinsichtlich Speicherplatz realisierbar ist.

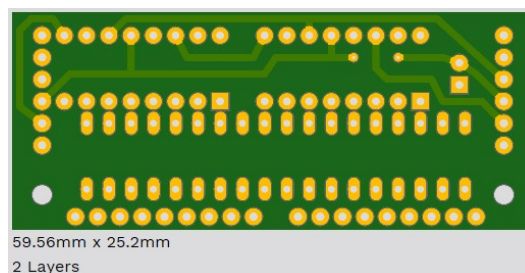
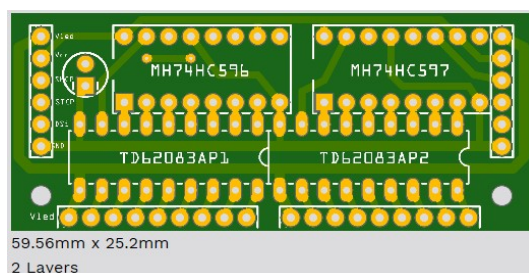
Die Platine vereint zwei Schieberegister 74HC595 mit jeweils einem ULN2803 zur Ausgangsverstärkung. Durch den 6-poligen Bus können mehrere Platinen hintereinander gesteckt werden und direkt an den ATtiny-Dekoder-3 oder den Nano-Dekoder-3 angeschlossen werden.

Die Bestückung erfolgt komplett mit konventionellen Bauteilen (keine SMD-Komponenten).

Fritzing-Zeichnung



Platinen-Layout



Freigabe-Link

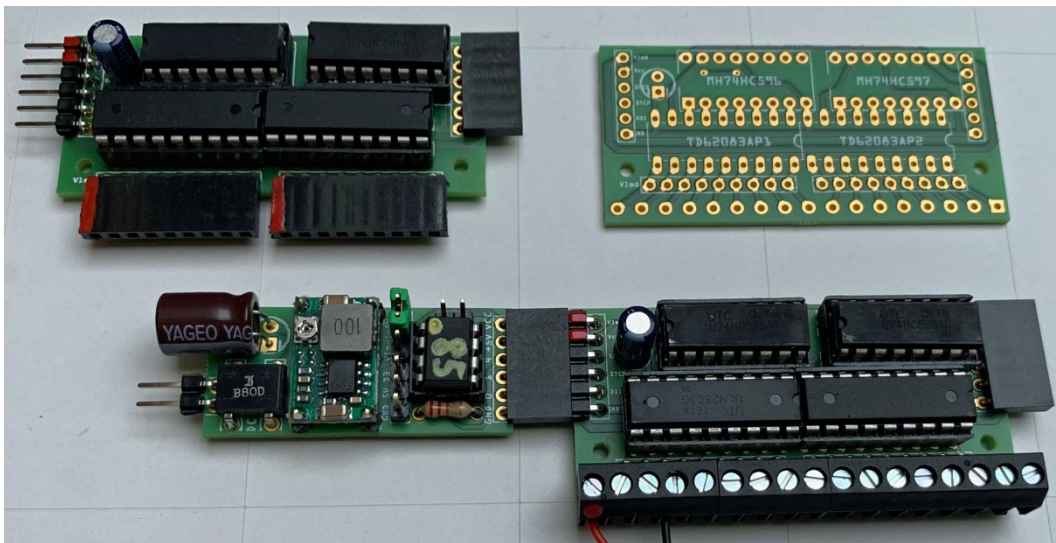
Siehe rev.2 dieser Platine.

Benötigte Bauteile

	z.B. Artikel-Nr. bei Reichelt Elektronik	
1 x Kondensator 1µF	RAD 1/63	0,02 €
2 x Schieberegister 74HC595		
2 x Darlington-Transistor-Array ULN2803	ULN 2803A	0,29 €
Optional: Stift-/Buchsenleisten RM 2,54mm (6-polig/9-polig)		
Optional: je 2x Stecksocket für 74HC595 und ULN2803 (16- und 18-polig)		

Hinweise zur Bestückung

Die Ausgänge können sowohl mit 9-poligen Stift- bzw. Buchsenleisten RM 2.54mm (auf der Abbildung oben links), als auch (ab rev.2) mit Schraubklemmen RM 3,5mm bestückt werden (auf der Abbildung unten rechts) – im zweiten Fall gibt es dann allerdings nur einen „+“-Anschluss für alle 16 Ausgänge:



Hinweise zu Anschluss und Betrieb

Schieberegister zweifach (rev.2, 06.12.2018)

Die zweite Version der Schieberegister-Platine unterscheidet sich von Version 1 nur durch die zusätzliche Möglichkeit der Montage von 3,5mm-Schraubklemmen für die Ausgänge.

Freigabe-Link

Unter <https://aisler.net/p/JQEXFPKS> kann diese Platine bei AISLER bestellt werden.



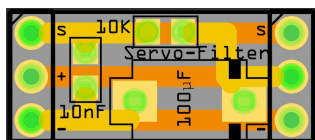
Servo-Filter (rev.5, 18.06.2018)

Zur Reduzierung unkontrollierter Bewegungen von Modellbau-Servos gibt es verschiedene Ansätze, um die unterschiedlichen Ursachen zu bekämpfen.

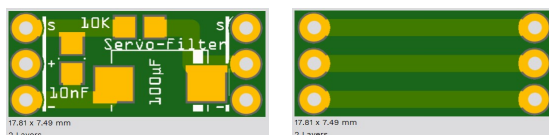
- Beim Einschalten liegt noch kein definiertes Signal an. Hier kann ein PullUp-Widerstand in der Größenordnung von ca. 10K zwischen der Signal-Leitung und +5V Abhilfe schaffen.
- Die relativ hohe Stromaufnahme beim Anlaufen des Servos führt zu Spannungseinbrüchen und damit zu Störungen der Elektronik. Hier kann eine Pufferung der Versorgungsspannung helfen.
- Bei längeren Leitungen kommt es zu Störimpulsen, die durch einen Kondensator zwischen Signal-Leitung und Masse herausgefiltert werden können.

Die Schaltung lässt sich mit „konventionellen“ Bauteilen problemlos auf einer Streifenrasterplatine aufbauen, benötigt dann aber relativ viel Platz. Die SMD-Variante lässt sich dagegen eventuell auch noch in Fahrzeugen unterbringen.

Fritzing-Zeichnung



Platinen-Layout



Freigabe-Link

Unter <https://aisler.net/p/MKDFUCES> kann diese Platine bei AISLER bestellt werden.

Benötigte Bauteile

	z.B. Artikel-Nr. bei Reichelt Elektronik	
1 x Widerstand 4,7K bis 10K, SMD 0805	RND 0805 1 10K	0,02 €
1 x Keramik-Kondensator 10nF, SMD 0805	X7R-G0805 10N	0,02 €
1 x Kondensator 100µF, SMD	T491D 100U 10 oder SMD TAN.100/6,3	0,32 €

Hinweise zur Bestückung

Achtung: Bei SMD-Tantal-Kondensatoren ist der Markierungstreifen die „+“-Markierung.

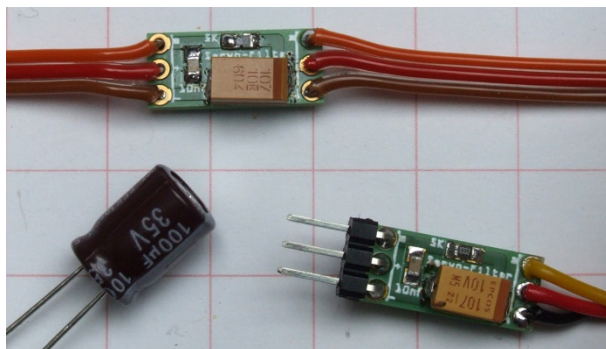


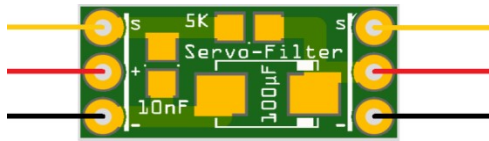
Abbildung zeigt rev.4

Hinweise zu Anschluss und zum Betrieb

Die Platine hat keinen „Eingang“ und „Ausgang“ – es ist also egal, ob das Servo links oder rechts angeschlossen ist.



Die Platine wird einfach in die Servo-Zuleitung einbunden. Dabei gilt (bezogen auf die Abbildung) „von unten nach oben“ Masse / 5V / Signal:



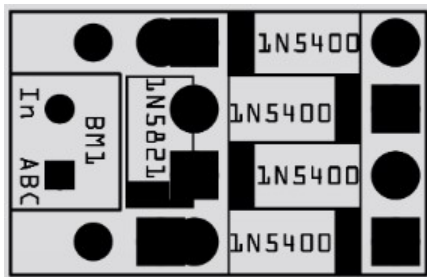


Bremsmodul Typ „BM1“ für Lenz ABC (rev.2, 23.05.2020)

Nachbau des Bremsmoduls vom Typ BM1 für das ABC-Bremssystem.

Fritzing-Zeichnung

Platinen-Layout



Freigabe-Link

Unter ... kann diese Platine bei AISLER bestellt werden.

Benötigte Bauteile

z.B. Artikel-Nr. bei Reichelt Elektronik

4 x Diode 1N5400 (3A)

1 x Diode 1N5821 (3A Schottky)

1 x Schraubklemme 2-fach RM 2,54

Hinweise zur Bestückung

Hinweise zu Anschluss und zum Betrieb