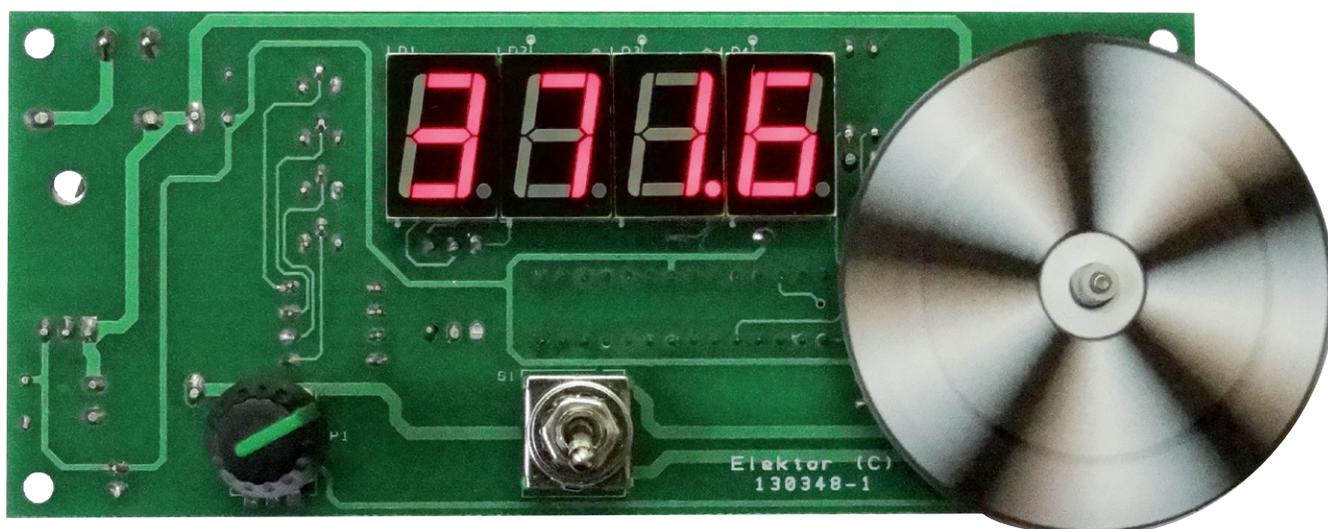


# Spulenwickler

## Dreht und zählt

Von Rolf Gerstendorf



Bei Induktivitäten für niederfrequente Anwendungen im Lang- und Mittelwellenbereich können leicht mehrere hundert Windungen zusammenkommen. Da wird manuelles Wickeln zur Qual. Dieser automatische Spulenwickler macht Schluss mit dem Elend. Er dreht den Spulenkörper im und gegen den Uhrzeigersinn und der eingebaute Zähler zeigt die Windungszahl aufs Zehntel exakt im Display an.

Der Spulenwickler soll, wie der Name verrät, Spulen wickeln und dabei die Anzahl der Umdrehungen sowie die Drehrichtung des Motors mit Hilfe einer an der Motorwelle befestigten schwarz/weißen Scheibe bestimmen. Die Scheibe ist in zehn Abschnitte aufgeteilt, abwechselnd weiß und schwarz gefärbt. Zwei Reflex-Optokoppler registrieren Umdrehung und Drehrichtung. Wenn der Motor im Uhrzeigersinn dreht, wird der Zählwert inkrementiert, dreht er gegen den Uhrzeigersinn, wird der Zählwert dekrementiert und das Resultat in einem Display dargestellt.

### Hardware

In der Schaltung in **Bild 1** lassen sich leicht fünf Abteilungen ausmachen: Ein Display, bestehend aus vier 7-Segment-Anzeigen, ein Drehrichtungs/Zählimpulsgenera-

tor, beide an einem Mikrocontroller angeschlossen, eine Motorsteuerung und eine Betriebsspannungstabilisierung.

Der Spulenwickler basiert auf der Intelligenz, die in dem Mikrocontroller ATmega328P (IC2) steckt. Der Controller verwendet das interne Taktsignal von 8 MHz. Die Hauptfunktion des Controllers ist es, die Anzahl der Umdrehungen der Motorwelle in Vorwärts- oder Rückwärtsrichtung zu zählen und vier 7-Segment-Displays anzusteuern. Der Maximalwert ist „999.9“.

Die Schaltung um die beiden Reflex-Optokoppler CNY-70 (IC3 und IC4) erzeugt die Zähl- und Richtungssignale für den Controller. Die Optokoppler sind direkt nebeneinander angebracht und beleuchten die schwarz/

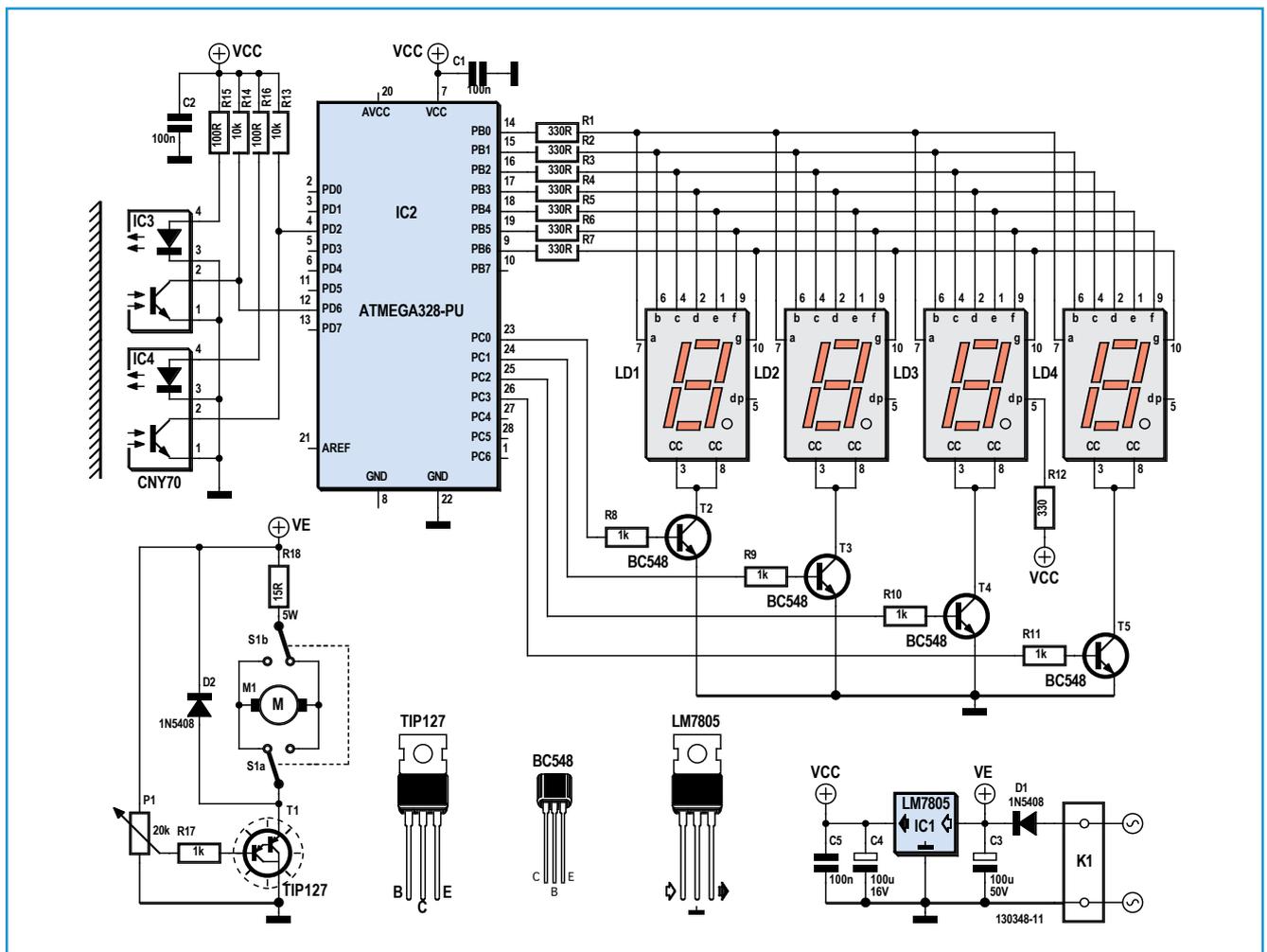
weiße Scheibe. Trifft das Licht der internen LED auf einen weißen Bereich der Scheibe, so wird es reflektiert, der korrespondierende Phototransistor schaltet durch und legt den angeschlossenen Mikrocontroller-Eingang auf Masse. Der schwarze Bereich der Scheibe „schluckt“ dagegen das Licht, der Phototransistor wird nicht beleuchtet und sperrt, so dass der entsprechende Controllereingang wegen des Pullup-Widerstands R13/R14 auf High geht.

Die Scheibe weist fünf weiße und fünf schwarze Bereiche auf, es werden also pro Umdrehung zehn Pegelwechsel erzeugt. Die Software erkennt einen solchen Pegelwechsel von IC4 an PD2 als Zählimpuls und wertet gleichzeitig den Zustand von IC3 (an PD6) aus. Sind beide Pegel gleich, wird der Zähler erhöht, sind sie unterschiedlich, wird der Zählerwert vermindert.

Das Display ist mit vier Halb Zoll-LED-Displays mit gemeinsamer Kathode aufgebaut. Die sieben Segmente von LD1...LD4 sind parallel an den Controllerausgängen PB0...PB6 angeschlossen. Die Pins PC0...PC3 treiben die gemeinsamen Kathoden der Displays über die Treibertransistoren T2...T5. Der Dezimalpunkt von LD3 ist über R12 fest mit der Versorgungsspannung verbunden, das Display hat also das Format „000.0“.

Der Spulendreher verwendet den kleinen, etwa 70 g leichten 12-V-Gleichspannungsmotor N2738 von Igarashi [1]. Dieser Motor ist in Modellbaukreisen sehr beliebt und deshalb gut und preiswert erhältlich. Bei einem Laststrom von 1,43 A (11,4 W) erreicht er eine Drehzahl von 13.000 U/min und seine höchste Effizienz von 66,7 %. Natürlich wickeln wir Spulen nicht mit dieser Geschwindigkeit (da

Bild 1. Schaltung des Spulendrehers.



würden die Finger qualmen!), die Werte zeigen aber, dass der Motor ausreichend Drehmoment (8,37 mNm) für unsere Applikation zur Verfügung stellt.

Die Motorsteuerung ist völlig unabhängig von der Mikrocontrollerschaltung und fällt unter die Kategorie „simpel“. Der Motor wird über einen Darlington-Transistor T1 angetrieben, an P1 steuert man durch Ändern des Basisstroms von T1 die Drehzahl des Motors. Stellen Sie an P1 die Drehzahl ein, die Ihnen behagt. Je dicker der Draht, desto langsamer sollte der Motor drehen. Da der Motor, wenn er blockiert, beinahe 10 A ziehen kann, ist ein Widerstand R18 zur Begrenzung des Motorstroms notwendig. Mit dem doppelpoligen Umschalter S1 (DPDT ON-OFF-ON) wird die Drehrichtung des Motors bestimmt. D2, eine 1N5408, ist die übliche Freilaufdiode mit einem Nennstrom von 3 A.

Die Schaltung erhält ihre Versorgungsspannung aus einem 12-V-DC-(Stecker-)Netzteil, das an K1 angeschlossen wird. D1 sorgt für einen Verpolschutz, C3 puffert die Eingangsspannung. Der Motor wird direkt mit dieser unstabilierten Spannung betrieben, während der Rest der Elektronik mit einer von IC1 auf +5 V heruntergeregelten Spannung versorgt wird.

**Software**

Die Software wurde in AVRstudio in der Programmiersprache C geschrieben. Sie kann sowohl als Sourcecode als auch in kompilierter Form kostenlos von [2] heruntergeladen werden. **Bild 2** zeigt die notwendigen Fuse-Einstellungen beim Programmieren in AVR-Studio.

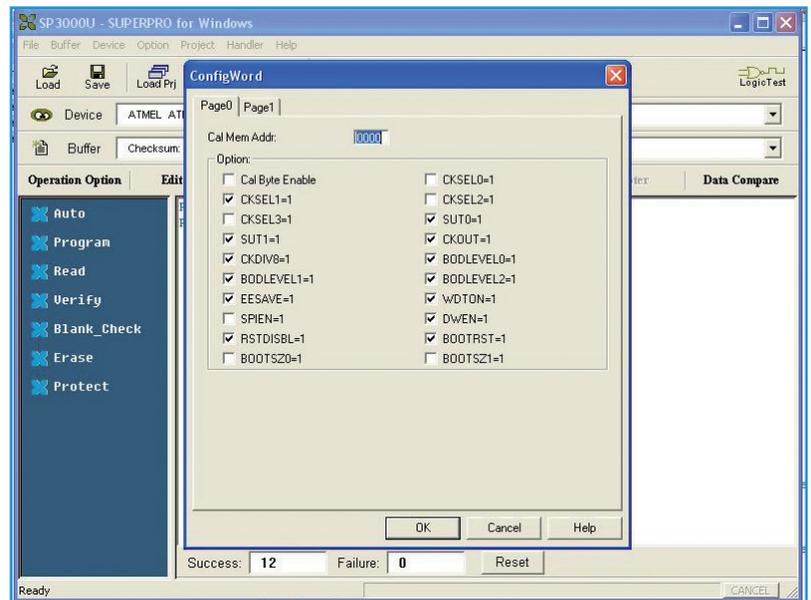
Das Hauptprogramm besitzt zwei Funktionen:

*Display\_seg-Funktion*

Diese Funktion nimmt den Wert von count als Argument, trennt den Wert in einzelne Ziffern auf und stellt diese auf den 7-Segment-Anzeigen dar.

*ISR0-Routine*

PD2 fungiert als Interrupt-Eingang. Bei jeder Statusänderung an PD2 des Mikrocontrollers wird die Programmsteuerung an diese Funktion übergeben. Sie prüft den Wert von PD6



und erhöht (PD6 ist gleich) oder verringert (PD6 ist ungleich) den Zählwert.

Bild 2. Fuse-Einstellungen bei der Programmierung.

*Hauptfunktion*

Diese Funktion stellt die Datenrichtung der Pins ein, initialisiert, ermöglicht die Interrupts und ruft die Display\_seg-Funktion auf.

**Mechanik**

Alle Bauteile – elektrische wie mechanische – sind auf der Platine (**Bild 3**) untergebracht, die mit der kostenlosen PCB-Software *Designspark PCB* [3] erstellt wurde. Das Layout finden Sie genau wie die Software auf der Projektseite [2].

Die Platine ist so gestaltet, dass sie gut in einem einfachen Gehäuse untergebracht werden kann. Für Poti P1, Schalter S1, die vier Displays (diese Bauteile befinden sich auf der Platinenoberseite) und die Motorwelle müssen Löcher in das lichtdichte Gehäuse gebohrt/ausgeschnitten werden. Die Displays werden nicht direkt auf die Platine gelötet, sondern in Buchsenleisten gesteckt.

Auch die beiden Optokoppler kommen auf die Oberseite. Alle anderen Bauteile gehören auf die Rückseite der doppelseitigen Platine. Transistor T1 muss mit einem guten Kühlkörper ausgestattet werden.

Die schwarz/weiße Scheibe wird in zehn Abschnitte unterteilt und gefärbt. Am besten machen Sie eine Kopie von **Bild 4**, dann

### Stückliste

**Widerstände:**

- R1...R7,R12 = 330 Ω
- R8...R11,R17 = 1 k
- R13,R14 = 10 k
- R15,R16 = 100 Ω
- R18 = 15 Ω, 5 W
- P1 = Poti 20 k

**Kondensatoren:**

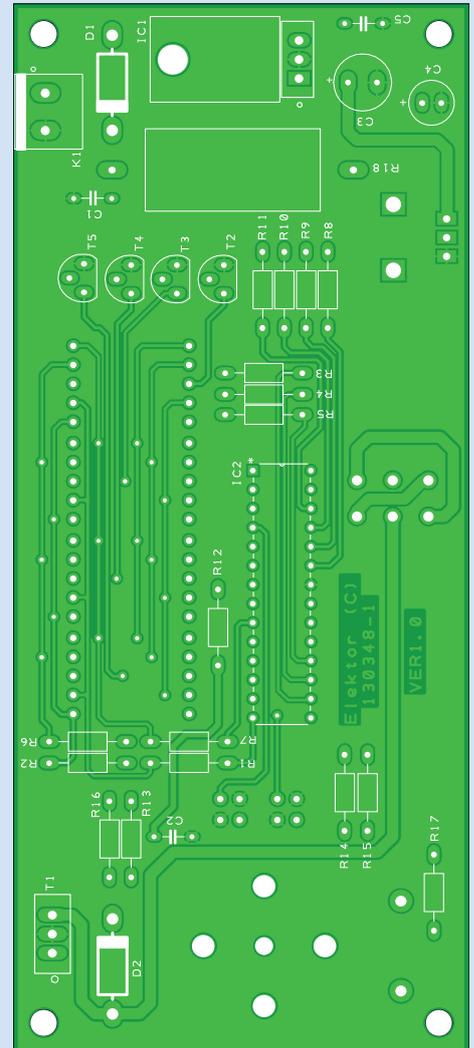
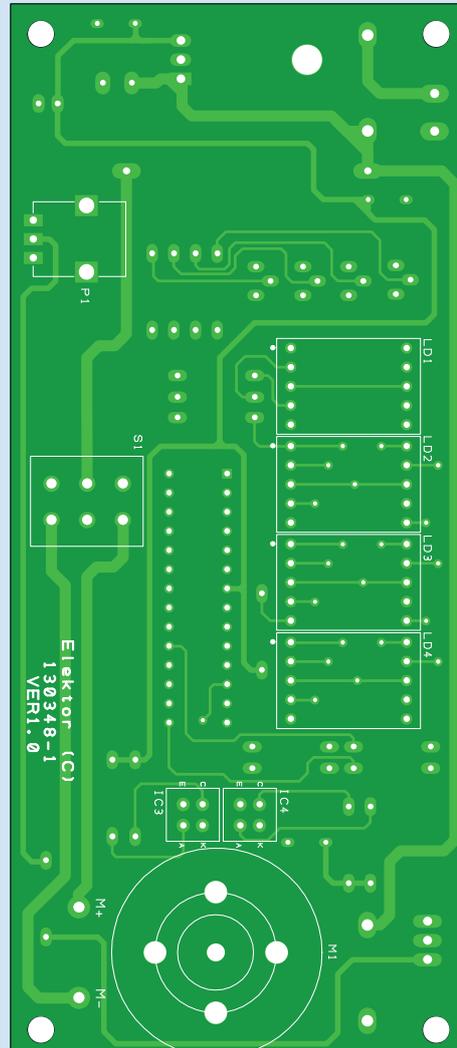
- C1,C2,C5 = 100 n
- C3 = 100 µ/50 V
- C4 = 100 µ/15 V

**Halbleiter:**

- D1,D2 = 1N5408
- LD1...LD4 = 7-Segment-Anzeige  
0,5" (gem. Kathode) z.B. King-  
bright SC05-11EWA
- T1 = TIP127
- T2...T5 = BC548
- IC1 = LM7805
- IC2 = ATmega328-PU (Atmel),  
programmiert (Elektor 130348-  
41 [2])
- IC3,IC4 = CNY70 (Vishay)

**Außerdem:**

- S1 = Kippschalter DPDT (TE  
Connectivity A203SYZQ04)
- K1 = 2-polige Platinenanschluss-  
klemme RM5
- Buchsenleisten 2,54 mm
- 12-V-Motor N2738-051-G-5 RS-  
385SH (Igarashi), bei Conrad  
244520
- Propeller-Nabe-Reely für  
2,3-mm-Welle (10583), bei  
Conrad 224235
- Platine Elektor 130348-1 [2]



haben Sie die Scheibe direkt in der richtigen Größe von 50 mm Durchmesser. Wenn Sie die Kopie (mittig, um Unwucht zu vermeiden) auf ein kreisförmig ausgeschnittenes Stückchen Platinenmaterial kleben, ist die Scheibe auch schön stabil. Der Durchmesser des zentralen Lochs ist abhängig von der Befestigungsart an der Motorwelle.

Der Motor selbst wird an der Platine an der im Layout bezeichneten Stelle mit zwei gewöhnlichen Schrauben à M3 x 0,5 x 5 mm befestigt. Sie dürfen nicht weiter als 3 mm in den Motorblock gedreht werden! Es gibt mehrere Möglichkeiten, die Scheibe und natürlich auch den Spulenkörper wackelfrei an der 2,3 mm durchmessenden Motorwelle zu befestigen [4]. Wir haben uns für die Variante mit einer Propellernabe entschieden. Die Scheibe kann

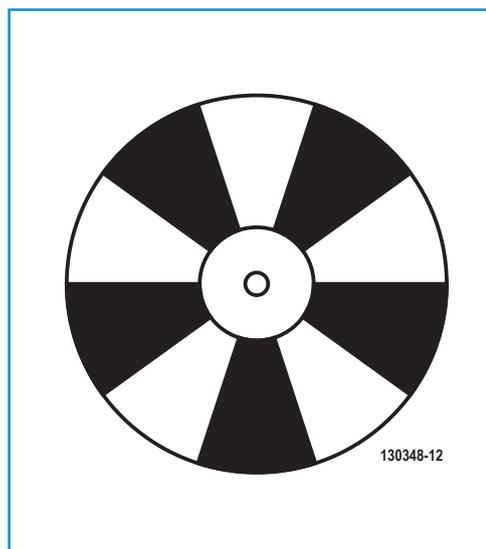


Bild 3. Auf der doppelseitigen Platine sitzen alle Bauteile inklusive Motor.

Bild 4. 1:1-Kopiervorlage für die Schwarzweiß-Scheibe.

dann direkt auf den Schaft geklebt werden. Welche Befestigungsart Sie aber auch immer bevorzugen, achten Sie darauf, dass zwischen Scheibe und Optokopplern ein Abstand von etwa 2 mm sichergestellt ist.

Wenn Sie den Motor nicht an der Platine befestigen, sondern zum Beispiel eine Bohrmaschine als Antrieb verwenden möchten, dürfen die Optokoppler natürlich nicht auf der Platine bestückt werden. Wie auch immer Ihre Lösung aussieht, Sie müssen zwei Dinge

beachten: Zum ersten muss zum Anschluss der Optokoppler an die Platine ein mehradriges abgeschirmtes Kabel verwendet werden, damit keine Störungen als Spikes zum Mikrocontroller vordringen und fehlerhafte Zählungen verursachen können. Und zum zweiten muss natürlich dafür gesorgt werden, dass kein Umgebungslicht auf die Optokoppler fallen kann.

(130348)

#### Weblinks

[1] [www.igusa.com/pages/motors/N2738.html](http://www.igusa.com/pages/motors/N2738.html)

[2] [www.elektor-magazine.de/post](http://www.elektor-magazine.de/post)

[3] <http://designshare.designspark.com/eng/projects/122/view/files>

[4] [www.rn-wissen.de/index.php](http://www.rn-wissen.de/index.php), Stichwort: Räder an der Motorwelle befestigen