



## Schritt für Schritt präzise steuern – I<sup>2</sup>C-Schrittmotorsteuerung iSMT

Schrittmotoren sind heute auch für den Hobbyelektroniker das Mittel der Wahl, wenn es darum geht, präzise steuerbare Antriebe für vielfältige Aufgaben zu realisieren – vom Modellbau über die Haustechnik bis hin zum Roboter- oder Maschinenbau. Mit der leistungsfähigen Schrittmotorsteuerung iSMT, die auf einem aktuellen Schrittmotor-Treiber-Interface basiert, stellen wir eine präzise Lösung vor, die bis zu 16 Mikroschritte pro Vollschritt ebenso beherrscht wie die sichere Steuerung in Bezug auf Beschleunigung, Drehgeschwindigkeit und Positionierung. Die Ansteuerung erfolgt über den I<sup>2</sup>C-Bus, der den Betrieb von bis zu 32 dieser Steuermodule ermöglicht.

### Faszination Schrittmotor

Der Schrittmotorantrieb ist für den Elektroniker eine Herausforderung, gilt es doch hier, die mechanischen Möglichkeiten eines Schrittmotors auf der Steuerungsseite voll auszureizen und zu beherrschen. Selbst für

den Hobbyelektroniker sind heute Schrittmotoren verfügbar, die eine Schrittweite von 0,9 Grad aufweisen, also schon im Vollschrittbetrieb pro Umdrehung 400 Positionen beherrschen. Dreht man den Rotor eines solchen Schrittmotors von Hand, so erkennt man

hier kaum noch die Einzelschritte. Eine elektronische Ansteuerung, die bis zu 16 Mikroschritte realisieren kann, macht aus jedem dieser mechanischen Einzelschritte 16 Mikroschritte, was dann zu einer extrem feinen Auflösung von 6400 Schritten je Umdrehung führt! Und da man über die elektronische Ansteuerung immer Kenntnis über die genaue Position der Achse des Motors hat, weiß man im Prinzip auch immer, in welcher Position sich das mit dem Motor bewegte Objekt genau befindet.

Woran denkt man bei einem Schrittmotor meist zuerst? Wohl an Roboter, egal in welcher Form sie uns begegnen, ob als Spielzeug,

Ansteuerbare Motoren:	2-Phasen-Schrittmotoren (4-, 6- oder 8-polig)
Max. Phasenstrom:	800 mA (pro Phase)
Positionsauflösung:	Teilung in Mikroschritte von $\frac{1}{2}$ , $\frac{1}{4}$ , $\frac{1}{8}$ , $\frac{1}{16}$ , interner 16-Bit-Positionszähler
Diagnose/Schutz:	Überstrom, Unterspannung, Übertemperatur, Kabelbruch, sensorfreie Blockadeerkennung, Schrittverlusterkennung
Vollschrittfrequenz:	bis 1 kHz
Konfigurierbar:	Min./max. Geschwindigkeit, Beschleunigung, Position, Lauf-, Haltestrom, Richtung, Schrittgröße
Schnittstelle/Bus:	I <sup>2</sup> C (bzw. TWI, wenn mit TMC223 bestückt)
Paralleler Betrieb:	bis zu 32 Steuerplatinen an einem Bus
Spannungsversorgung:	9–29 V <sub>DC</sub>
Stromaufnahme:	max. 1600 mA
Bohrung (Montageraster):	M3 (26 x 26 mm und 31 x 31 mm), M4 (41,74 x 41,74 mm)
Abmessung (B x H x T):	49 x 21 x 49 mm

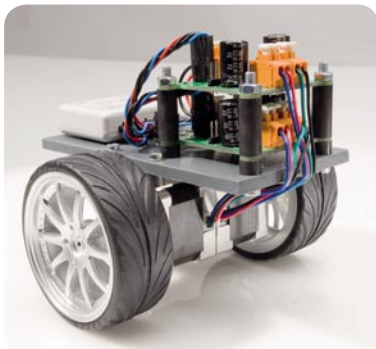


Bild 1: Anwendungsbeispiel: ein kleines Roboterfahrzeug zum Experimentieren mit Schrittmotoren, das durch pfiffige Programmierung sogar zum selbstbalancierenden Modell werden kann

als Fahrzeug, in der Industrie oder der Forschung. Besonders beliebt ist die Robotertechnik bei Hobbyelektronikern wohl als Fahrzeug, zunehmend erobert sich aber auch der Hobbyist andere Gebiete, wenn man nur einmal an die vielen Arten der Mini-Roboter denkt, die sich auf Erfindermessen oder Roboterfußballturnieren tummeln. Ein einfaches Roboterfahrzeug ist in Bild 1 zu sehen, das – ähnlich den sogenannten Turtles – mit zwei Schrittmotoren, zwei der hier vorgestellten iSMT-Steuerungen, der I<sup>2</sup>C-Beschleunigungssensorplatte 3D-BS aus dem ELVjournal 3/2010 und einem USB-I<sup>2</sup>C-Interface von ELV ausgestattet ist. Mit einem solchen Aufbau kann man bereits Fahrsteuerungen realisieren. Mit den Daten aus dem 3-Achsen-Beschleunigungssensor kann die Lage des Fahrzeugs jederzeit überprüft werden. Das koordinierte Zusammenwirken der einzelnen Komponenten ist dann nur noch eine Herausforderung an das Programmieren einer Steuerung auf einem PC. Will man ein völlig autonomes Fahrzeug realisieren, erfolgt die Steuerung idealerweise über ein Mikrocontrollerboard, wozu sich z. B. das in dieser Ausgabe vorgestellte Arduino-Board hervorragend eignet, das sogar mit GPS und Display ausgestattet werden kann und zudem über eine I<sup>2</sup>C-Schnittstelle verfügt. Wer zu diesem Thema Anregungen benötigt, sollte einmal auf [1] oder in den zum o. a. Arduino-Board genannten Quellen stöbern. Auch

die hier vorgestellte Kombination mit dem USB-I<sup>2</sup>C-Interface ist aufgrund des im Interface speicherbaren Makros bereits als autark arbeitende, sehr einfache Steuerungslösung einsetzbar.

Eine große Faszination, vor allem für denjenigen, der auch eine Affinität zur Mechanik hat, üben Schrittmotoren in selbst gebauten Maschinen aus, wie z. B. Zeichenplotter, Drehbänke oder CNC-Fräsen, wie sie zur Kunststoff- oder Leiterplatten-Bearbeitung eingesetzt werden. Hier sorgen in bis zu vier Achsen (kräftige) Schrittmotoren für präzise und genauestens

definierte Positionierung von Stiften, Bohrern, Fräsen. Wer sich die mechanische Realisierung zutraut, kann hier für vergleichsweise wenig Geld, aber auf jeden Fall mit dem Erfolgserlebnis „selbstgebaut“ z. B. eine CNC-Fräse selbst bauen. Wird deren Steuerung dann mit Fräs- und Bohrdaten des Platinenlayoutprogramms oder eines 3D-CAD-Programms gefüttert, dann arbeitet solch ein Gerät präzise Leiterbahnkonturen heraus, bohrt Löcher oder fräst 3D-Modelle. Auf [2] sind Interessenten zum Thema CNC-Fräse genau an der richtigen Stelle.

Wer übrigens nicht gleich teuer einsteigen und sich dem Thema erst einmal experimentell nähern will, findet ein sehr preiswertes Angebot für einen einfachen Schrittmotor (Bild 2). Auf zwei deutlich leistungsfähigere Motoren von ELV mit 200 Vollschritten werden wir noch im Verlauf dieses Artikels eingehen.

## Wie funktioniert er?

Ein Schrittmotor ist vom grundlegenden Arbeitsprinzip ein Elektromotor mit einem fest stehenden Stator und einem Rotor. Letzterer besteht

je nach Schrittmotor-Typ aus einem Weicheisenkern mit einer Zahnung (Reluktanz-Schrittmotor) oder einem Permanentmagneten mit radialer Magnetisierung. Da beide Prinzipien systembedingte Nachteile haben (der erste hat nicht die bei der Anwendung gewünschte Rastung, der zweite begrenzt die Quantelung der Schrittzahl mechanisch), hat man sie im heute am meisten verbreiteten Hybrid-Schrittmotor kombiniert. Dieser (in Bild 3 zerlegt zu sehen) verfügt über einen Rotor, der sowohl die beim Permanentmagnet-Schrittmotor beschriebene Magnetanordnung als auch den beim Reluktanz-Schrittmotor erwähnten gezahnten Weicheisenkern enthält. Beide sind, wie in Bild 3 gut zu sehen, versetzt angeordnet. Dies führt zu einer feineren Teilung und einem höheren Drehmoment des Motors.

Wir wollen uns anhand von Bild 4 die Funktion des Motors näher ansehen. Wie gesagt, besteht der Motor aus einem Rotor und einer geraden Anzahl von Spulen, im Beispiel von Bild 3 sind dies 8, die elektrisch angesteuert ein Magnetfeld im Stator hervorrufen. Die Spulen sind so angeordnet, dass sie sich paarweise genau gegenüber liegen (Polpaar).

Je nach gerade an den Spulen anliegender Stromrichtung erzeugen diese ein Magnetfeld mit Nord- oder Südpolung, das auf die Permanentmagnete des Rotors wirkt. Durch das Abstoßen gleichnamiger Magnetfelder und das Anziehen unterschiedlich gepolter Magnetfelder wird eine Drehbewegung erzeugt. Durch die mögliche Steuerung der Polpaare wird genau definiert, wie weit und wie schnell sich der Rotor zu drehen hat.

Bei dem in Bild 4 in vereinfachter Form mit 2 Polpaaren abgebildeten Motor können also im Vollschrittbetrieb 4 Schritte je Umdrehung realisiert werden. Bei einer elektronischen Schrittteilung (Mikroschrittbetrieb) werden die Schrittwinkel verkleinert, indem im Wechsel immer mehrere Spulen mit unterschiedlichen Strömen beaufschlagt werden. Je feiner diese Stromabstufung vorgenommen wird, desto höher wird die erreichbare Anzahl der Schrittpositionen, wie Bild 5 veranschaulicht. Diese Auflösung ist eine Frage des Aufwands auf der Ansteuerungsseite. Unsere iSMT-Endstufe erreicht bis zu 16 Mikroschritte.

Eine weitere wesentliche Eigenschaft des Schrittmotors ist die Art der Ansteuerung. Unterschieden wird zwischen der unipolaren Ansteuerung, für die vier Motorwicklungen nötig sind, und der bipolaren An-

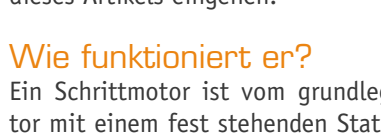


Bild 2: Preiswerter Motor für erste Versuche – Unipolar-Schrittmotor mit 96 Schritten pro Umdrehung (Best.-Nr.: 68-011-04)

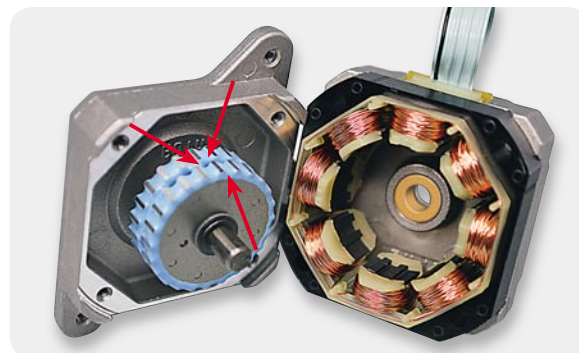


Bild 3: Der Aufbau eines bipolaren Hybrid-Schrittmotors. Man erkennt hier deutlich die typisch versetzten Magnet-Verzahnungen auf dem Rotor.

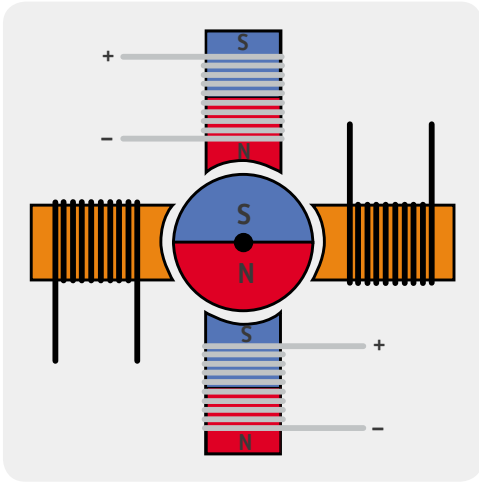


Bild 4: Der stark vereinfachte schematische Aufbau eines bipolaren Schrittmotors

steuerung, bei der zwei Wicklungen genügen. Die Bezeichnung „bipolar“ stammt vom Umpolen der Wicklungsströme, wodurch der Stator in Bewegung versetzt wird. Bei der unipolaren Ansteuerung wird der Strom nicht umgepolt, sondern die Spannung an vier Wicklungen wird im Wechsel ein- und ausgeschaltet. Zwei Wicklungen sind dabei ein- und zwei sind ausgeschaltet, wodurch sich das Drehmoment im Gegensatz zur bipolaren Ansteuerung verringert. Die vier Wicklungen können wie in Bild 6 C und 6 D getrennt oder wie in Bild 6 B über Mittelanzapfungen realisiert sein. Durch den Abgriff ist es möglich, jeweils nur eine halbe Spule mit Strom zu versorgen. Es wird einfach der Mittenanschluss auf Masse und an den Wicklungsenden nacheinander Spannung angelegt. So ergibt sich ein wanderndes Magnetfeld, das den Rotor in Bewegung versetzt. Gegenüber dem bipolar angesteuerten Schrittmotor lässt sich hier allerdings kein so hohes Drehmoment erzeugen, da ja nur jeweils die halbe Wicklung genutzt wird und daher das erzeugte Magnetfeld nur halb so hoch ist wie das des bipolar angesteuerten Motors. Deshalb steuert die iSMT Motoren auch bipolar an.

Übliche Schrittmotoren haben 4, 6 oder 8 Anschlüsse. Wie in Bild 6 gezeigt, lassen sich alle diese Varianten mit der iSMT ansteuern. Für den Anschluss sind hier die Motoren mit zwei Wicklungen (Bild 6 A und B) und mit vier Wicklungen (Bild 6 C und 6 D) zu unterscheiden. Motoren mit vier Wicklungen lassen sich sowohl bipolar als auch unipolar ansteuern. Für die bipolare Ansteuerung müssen je zwei Wicklungen entweder parallel (Skizze C) oder seriell (Skizze D) zusammengeschlossen werden.

Bleibt schließlich noch zu besprechen, in welcher Weise die Motorsteuerung, die Drehzahl und das Drehmoment verändert werden können. Die iSMT verwendet hier die Stromregelung, genauer gesagt eine Konstantstrom-Regelung. Dabei begrenzt ein Regler den Strom durch die Spulen des Motors durch Vergleich mit einem Referenzwert. Wird der Strom zu hoch, wird er kurz abgeschaltet. Dies erfolgt mit einer hohen Taktfrequenz. Dies ist die effektivste Regelungsmethode und durch die vollständige Integration im Treiberchip auch keine Aufwandsfrage mehr.

### Die Schrittmotorsteuerung iSMT

Mit der Schrittmotorsteuerung iSMT steht eine universelle und hochmoderne Steuerung mit allen Features zur Verfügung, die man heute von einer solchen verlangt. Sie ist für bipolare Hybrid-Schrittmotoren mit einem Wicklungsstrom von bis zu 0,8 A und damit bereits für etwas kräftigere Motoren geeignet.

Die iSMT ist eine kompakte Schrittmotorsteuerung (nur 49 x 21 x 49 mm) mit Stromregelung und Po-

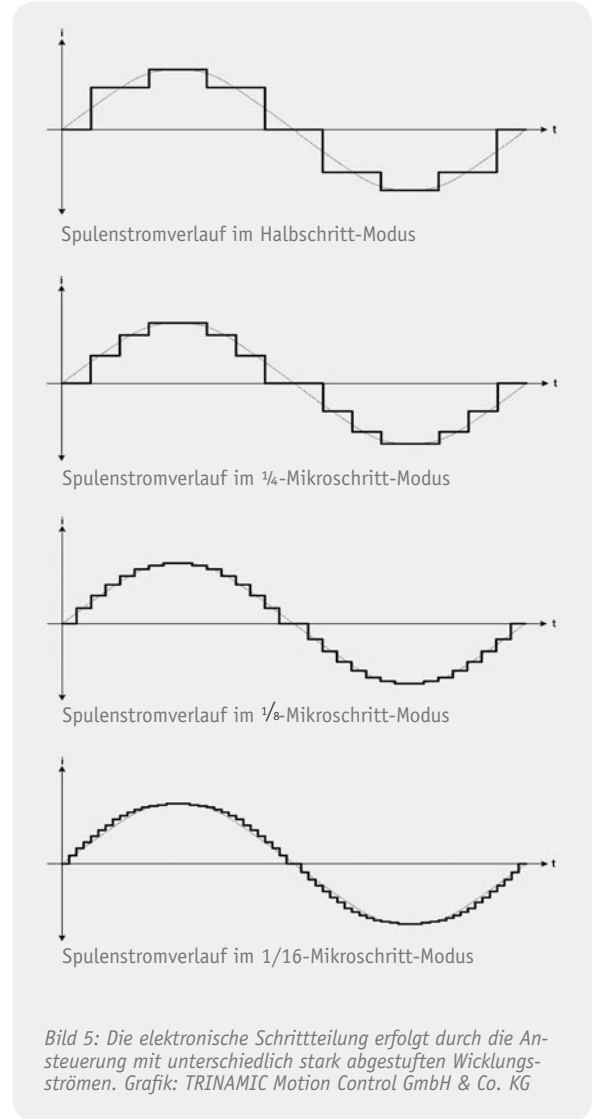


Bild 5: Die elektronische Schrittteilung erfolgt durch die Ansteuerung mit unterschiedlich stark abgestuften Wicklungsströmen. Grafik: TRINAMIC Motion Control GmbH & Co. KG

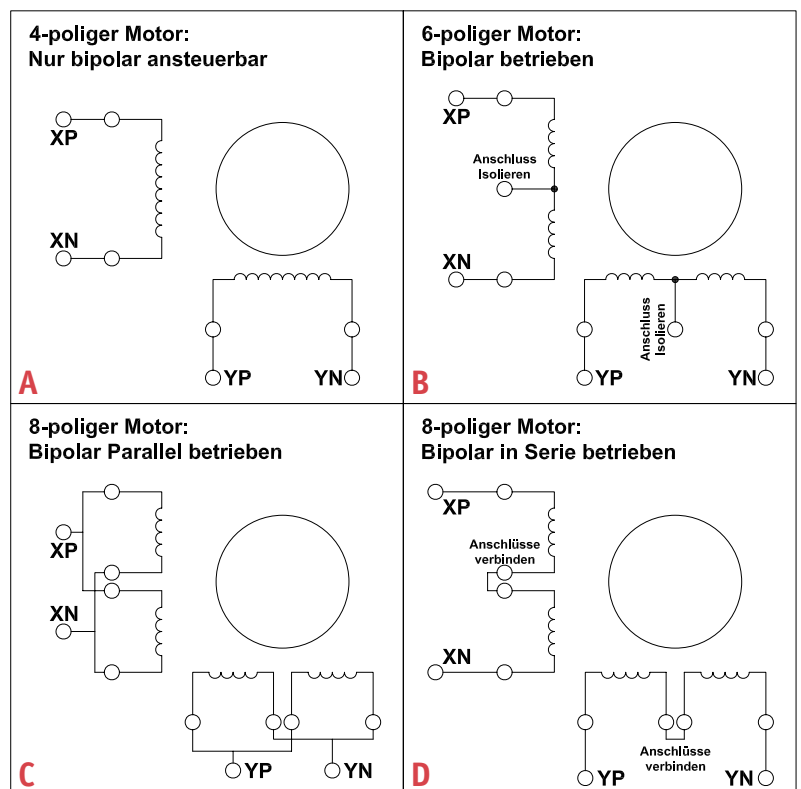


Bild 6: Anschlussmöglichkeiten von 4-, 6- und 8-poligen Schrittmotoren an die iSMT



Bild 7: Einfachste PC-Ansteuerung mit dem USB-I2C-Interface als Bindeglied zum iSMT

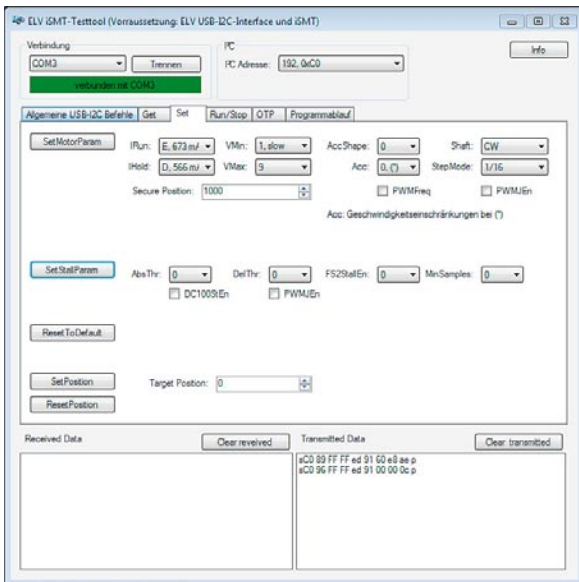


Bild 8: Über die per Download verfügbare PC-Software ist die einfache Konfiguration und Ansteuerung der iSMT über das USB-I2C-Interface möglich.

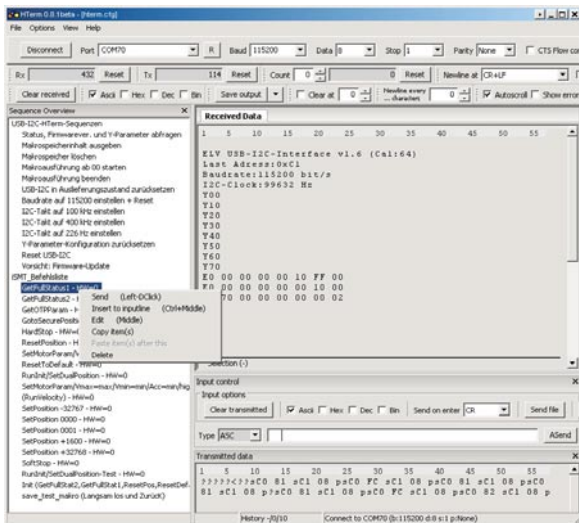


Bild 9: HTerm in Aktion. Links die Liste der verfügbaren iSMT-Befehle, die auf einen Doppelklick hin über das USB-I2C-Interface zur iSMT gesendet werden können.

sitionszähler auf Basis des vollintegrierten Treiberbausteins TRINAMIC TMC223 oder des baugleichen AMIS-30624. Sie ist sehr einfach einsetz- und konfigurierbar durch direkte Befehlssteuerung und Konfiguration über den zweiadrigen Standardbus I<sup>2</sup>C bzw. TWI und besitzt umfangreiche Diagnose- und Schutzfunktionen wie:

- Überstromschutz mit Signalisierung (mit GetFullStatus1 auslesbares Flag) getrennt für beide Wicklungen – auch kurzgeschlossene (oder falsch angeschlossene) Motoranschlussleitungen führen nicht zum Defekt der Schaltung
- Unterspannung mit Signalisierung (mit GetFullStatus1 auslesbares Flag) und Motor-Stopp, wenn Vbb < 8 V
- Übertemperatur mit mehrstufiger Signalisierung (mit GetFullStatus1 auslesbares Flag) und gesteuerter Motorabschaltung
- Untertemperatur-Signalisierung (mit GetFullStatus1 auslesbares Flag)
- Kabelbruch-Erkennung (bzw. auch falsch angeschlossener Motor oder aus Buchse herausgerutschtes Kabel)
- Sensorfreie Blockadeerkennung („stall detection») – wenn Motor ans mechanische Ende der Rotationsbewegung fährt, stoppt er, was eine leise, sehr genaue Positionskalibrierung während eines Referenzlaufs ermöglicht
- Schrittverlusterkennung – z. B. wenn die zu bewegende Masse zu träge für die vorgegebene Bewegung ist bzw. das Drehmoment des verwendeten Motors zu gering. Um Schrittverluste zu vermeiden, sollten zum einen hochwertige Hybrid-Schrittmotoren mit möglichst großem Drehmoment und zum anderen ein Sanftanlauf mit angemessener Beschleunigung verwendet werden.

Schraubfreie Anschlussklemmen für Motor und Spannungsversorgung ermöglichen eine schnelle und einfache Verkabelung. Durch den weiten Einsatzspannungsbereich von 9 bis 29 V ist nur eine gemeinsame Spannung für den Controller und den Motor notwendig, damit vereinfacht sich die Spannungsversorgung.

Im Prinzip ist es dank der Stall-Detection und der Schrittverlusterkennung nicht einmal notwendig, eine separate Positionserkennung für den Betrieb von Schrittmotoren installieren zu müssen. Ist dies dennoch einmal erforderlich, kann eine bestimmte Position per Positionsschalter (z. B. Endschalter) über einen frei verwendbaren Eingang (SWI am Jumper/Stiftleiste J 2) erfasst und ausgewertet werden.

Die kompakte Platine der iSMT kann direkt auf der Rückseite von Schrittmotoren befestigt werden. Dafür sind bereits Montagebohrungen in drei gängigen Rastern vorhanden: 26 x 26 mm und 31 x 31 mm mit 3-mm-Bohrungen sowie 41,74 x 41,74 mm mit Bohrungen für M4-Schrauben. Zur Montage liegen dem Bausatz bereits Schrauben, Abstandsrollchen und 2 Winkel (zur seitlichen Montage) bei. Diese Art der Montage auf der Motorrückseite hat den großen Vorteil, dass die Leitungen zum Schrittmotor sehr kurz sind, Störausstrahlungen so weitgehend unterdrückt und Spannungsverluste vermieden werden.

Über einen +5-V-Ausgang an ST 1 kann eine externe Schaltung, z. B. die für den I<sup>2</sup>C-Bus notwendigen Pull-up-Widerstände oder ein Mikrocontroller, mit bis zu 10 mA direkt von der iSMT versorgt werden (dafür ist die Sicherheits-Lötbrücke J 3 zu schließen).

### Einfache Steuerung über I<sup>2</sup>C

Durch die Steuerung über den I<sup>2</sup>C-Bus und die Adressiermöglichkeit (1 Bit der Adressierung erfolgt über Jumper J 1 und 4 Bit über die Konfiguration per Befehl) ist der parallele Betrieb von bis zu 32 iSMT-Steuerungen an einem Bus möglich. Damit vereinfacht sich auch die Verkabelung einer Steuerung mit mehreren Schrittmotoren. Der Bus unterstützt den I<sup>2</sup>C-Fast-Mode mit einer Taktrate bis 350 kHz.

Die iSMT ist direkt an das USB-I2C-Interface von ELV (Bild 7) anschließbar und damit sofort ohne Voraussetzung von Programmierkenntnissen einsetzbar. Denn dank der extrem einfachen Anwendung mit nur wenigen Befehlen und der Fähigkeit, im USB-I2C-Interface ein selbstlaufendes Makro speichern zu können, ist mit diesen zwei Bausteinen bereits

eine autark ohne PC laufende Schrittmotorsteuerung realisierbar. Auf der Produktseite zum USB-I2C-Interface [4] findet man auch die umfangreiche Dokumentation zum USB-I2C-Interface.

### Top-Support

Ein nicht zu unterschätzender Vorteil des verwendeten Treiberbausteins ist, dass sowohl AMIS als auch TRINAMIC äußerst umfangreiche Informationen zur Verwendung und Ansteuerung ihrer beiden fast identischen Treiber bereitstellen. Es gibt umfangreiche und sehr gute Datenblätter [5, 6] und sogar ein deutschsprachiges Forum und ein FAQ-Dokument vom Hersteller Trinamic zum TMC223 [6]. Ebenfalls von Trinamic gibt es unter [6] eine frei verfügbare und frei verwendbare C-Bibliothek zur Ansteuerung des TMC223 über den TWI-Bus für AVR-Mikrocontroller.

Um einen schnellen Einstieg in die Anwendung der iSMT zu gewährleisten, stehen auf der iSMT-Produktseite [7] zwei Software-Werkzeuge zum kostenfreien Download bereit. Das erste ist ein eigens entwickeltes PC-Tool (Bild 8), mit dessen Hilfe die iSMT über das USB-I2C-Interface einfach und sofort in Betrieb genommen und getestet werden kann. Dies ist sehr hilfreich zum Verständnis der Funktionsweise des Treiberchips, zeigt dessen umfangreiche Statusmeldungen, übernimmt die Steuerung des USB-I2C-Interface und erlaubt eine Steuerungssequenz zu erstellen, die direkt vom Testtool ausführbar ist. Um eigene Programme auf dem iSMT-Tool aufzubauen oder um dies Tool den eigenen Ansprüchen anpassen zu können, steht der komplette in C# geschriebene Source-Code des iSMT-Tools als kostenfreier Download zur Verfügung.

Eine genauere Einführung in die Funktionsweise des Programms findet sich in Teil 2 dieses Artikels.

Das zweite Werkzeug ist eine HTerm-Sequenz-Datei mit einer Befehlsliste zur Ansteuerung der iSMT über das USB-I2C-Interface. Das unter [8] kostenfrei erhältliche Terminalprogramm ist für diesen Zweck sehr empfehlenswert und bietet ab Version 0.8.0 die Möglichkeit, sogenannte Sequenz-Dateien, in denen ganze Befehls-Sequenzen gespeichert sind, zu erstellen, sie zu laden, zu speichern und durch einen Doppelklick sehr einfach zu senden. Die komplette Bedienung ist in der ausführlichen Dokumentation zum USB-I2C-Interface erläutert, an dieser Stelle wollen wir lediglich das Beispiel zur Schrittmotorsteuerung in Bild 9 näher betrachten.


Die Befehle der Schrittmotorsteuerung können im Sequenz-Fenster von HTerm durch Doppelklicken über das (angeschlossene und richtig verbundene) USB-I2C-Interface zur iSMT gesendet werden. Deren Antworten erscheinen im Terminalfenster oben. Sämtliche Befehlssequenzen der Schrittmotorsteuerung hier zu erläutern, würde den Umfang des Artikels sprengen, sie sind in den wirklich ausgezeichneten Datenblättern der Schrittmotor-Treiber ausführlich erläutert. Ohnehin kommt man um

das Studium dieser Datenblätter kaum herum, sollen sich über die Grundbefehle hinaus alle Funktionen, Befehle, Meldungen erschließen. Das o. a. PC-Tool ermöglicht zwar den ersten Einstieg auch ohne dieses Studium, jedoch muss man die genaue Bedeutung von Befehlen und Meldungen im Datenblatt nachschlagen. Insbesondere bei der Programmierung der OTP-Register (One-Time-Programmierung – nur einmal programmierbar, siehe Kapitel 5.2.1 im TMC223-Datenblatt) ist einiges zu beachten

Noch ein Hinweis am Schluss dieses Abschnitts: Die iSMT wird ab Werk, je nach Chip-Liefersituation, entweder mit dem TMC223 oder dem AMIS30624 bestückt. Da die Chips kompatibel sind (es scheint nach ausführlichem Vergleich beider Datenblätter lediglich eine lizenzrechtlich bedingte Unterscheidung in der Bezeichnung des I<sup>2</sup>C-Busses zu geben), verwendet man am besten beide Datenblätter, die sich gut ergänzen. Etwas ausführlicher und übersichtlicher ist dabei das Datenblatt der Firma AMIS/ON Semiconductors [5].

### Neue leistungsstarke Schrittmotoren

Zusätzlich zur iSMT haben wir zwei hochwertige, bipolare Hybrid-Schrittmotoren von MotionKing in das ELV-Programm aufgenommen, zu finden als Zubehörverweis unter [7]. Beides sind sehr leistungsstarke Motoren mit 1,8°-Schrittwinkel. Der 14HY3402 ist ein sehr kompakter und leichter Motor, der besonders für schnell arbeitende Antriebe ohne größere Last geeignet ist. Auf größere Lasten ist das etwas größere und deutlich kräftigere Modell 23HS4412 spezialisiert, das über ein hohes Haltemoment (die Kraft, die die Motorachse im Stillstand auf der Position hält) und ein höheres Drehmoment verfügt. Übrigens, 1,8° Drehwinkel bedeutet, dass der Motor 200 Vollschritte für eine Umdrehung realisiert – und bei 16facher Schrittteilung 3200 Schritte! Damit sind in Zusammenarbeit mit der iSMT sehr präzise Steuerungen möglich.

In Teil 2 dieses Artikels werden neben der Schaltungsbeschreibung und dem Nachbau der iSMT auch die Inbetriebnahme und die ersten Schritte mit der Test-Software ausführlich beschrieben. 



### Weitere Infos:

- [1] Ausführliche Infos und Forum zur Realisierung von Robotersteuerungen und zu Schrittmotoren: [www.roboternetz.de](http://www.roboternetz.de)
- [2] Infos und Forum zum Bau von CNC-Maschinen: [www.cncecke.de](http://www.cncecke.de)
- [3] Preiswerter Schrittmotor bei ELV für erste Versuche: webcode: #1158
- [4] Produktseite und alle Infos zum USB-I2C-Interface: webcode: #1159
- [5] Datenblatt und weitere Infos zum AMIS-30624: [www.onsemi.com/PowerSolutions/product.do?id=AMIS30624C6244G](http://www.onsemi.com/PowerSolutions/product.do?id=AMIS30624C6244G)
- [6] Datenblatt, AVR-C-Library, TMC223-FAQ und weitere Infos zum Trinamic TMC223: [www.trinamic.com/tmc/render.php?sess\\_pid=394](http://www.trinamic.com/tmc/render.php?sess_pid=394)  
Deutschsprachiges Forum zum Trinamic TMC223: [www.trinamic.com/ttdg/viewforum.php?f=16](http://www.trinamic.com/ttdg/viewforum.php?f=16)
- [7] iSMT-Produktseite: [www.elv.de](http://www.elv.de), bitte die Bestellnummer 92720 im Suchfeld eingeben
- [8] Das Terminalprogramm HTerm: [www.der-hammer.info/terminal](http://www.der-hammer.info/terminal)
- [9] Grundlagen, Tipps, ein Forum und viele Beispiele zu AVR-Programmierung: [www.mikrocontroller.net/articles/AVR-GCC-Tutorial](http://www.mikrocontroller.net/articles/AVR-GCC-Tutorial)