



Schritt für Schritt präzise steuern – I²C-Schrittmotorsteuerung iSMT Teil 2

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 I²C-Bus, der den Betrieb von bis zu 32 dieser Steuermodule ermöglicht.

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

Schaltungsbeschreibung

Die Beschaltung des TMC223 bzw. des AMIS-30624 in der Schaltung des iSMT (Bild 10) entspricht weitestgehend den Empfehlungen in den Datenblättern. Durch die Realisierung der Stiftleisten J 1 und J 2, der Lötbrücke J 3 und der optionalen Pull-up-Widerstände R 1 und R 2 lassen sich wirklich alle Features der Schrittmotortreiber individuell nutzen.

Auf der Versorgungsseite (KL 1) wird der Treiberbaustein IC 1 durch die 2-A-SMD-Sicherung SI 1 und die Schottky-Power-Diode D 1 geschützt, während ausgangsseitig (KL 2 und KL 3) der Treiber selbst für ein hohes Maß an Sicherheit sorgt.

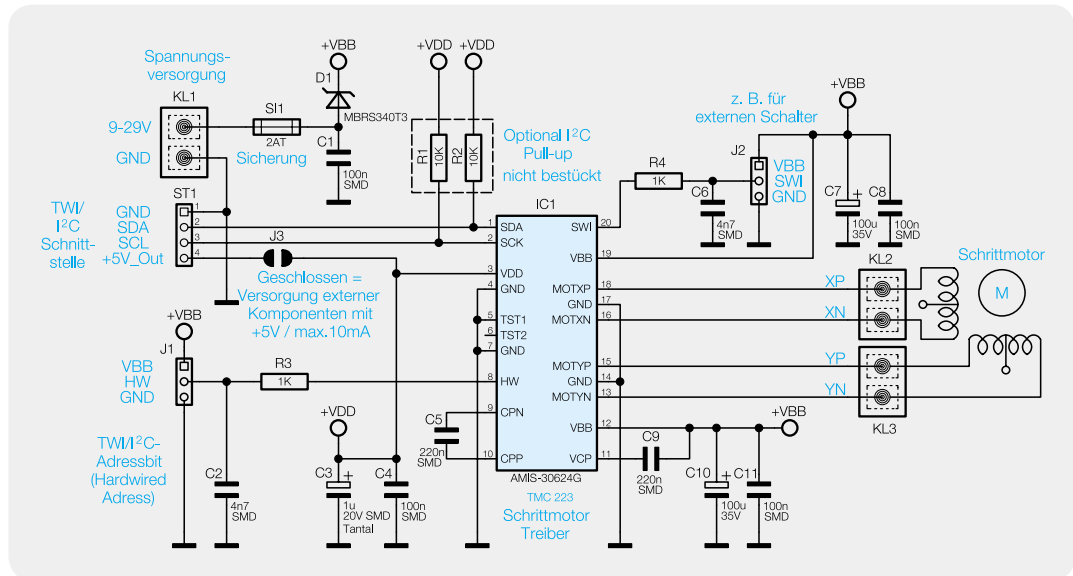
Über die Stiftleiste ST 1 wird der iSMT an einen TWI-/I²C-Bus angeschlossen. Verfügen die angeschlossenen Busleitungen SDA und SCL nicht bereits über die erforderlichen Pull-up-Widerstände, wie es z. B. beim USB-I²C-Interface der Fall ist, so können R 1 und R 2 nachträglich bestückt werden. In den meisten Fällen sollten dafür dann 10-k Ω -Widerstände (Bauform 0805) die richtige Wahl sein.

Ebenfalls über ST 1 kann die von IC 1 bereitgestellte +5-V-Spannung auch extern genutzt werden, wobei über ST 1 nur maximal 10 mA entnommen werden dürfen. Sicherheits halber, damit über diesen Anschluss nicht etwa fälschlicherweise 5 V eingespeist werden, muss vor der Verwendung der Spannung die Lötbrücke J 3 erst geschlossen werden.

Nachbau und Montage

Wie bei aktuellen ELV-Bausätzen üblich, sind auch beim iSMT alle SMD-Bauteile bereits werkseitig bestückt. Zum Einlöten bleiben lediglich die Jumper J 1, J 2, die Stiftleiste ST 1, die Klemmen KL 1 bis KL 3 und die Elkos C 7 und C 10. Bei den Elkos ist unbedingt auf die richtige Polung zu achten, da diese sonst explodieren können. Am Bauteil selber ist der Minuspol mit einer breiten Markierung gekennzeichnet, während auf der Platine der Pluspol gekennzeichnet ist. Als Letztes sollte der Adressjumper J 1 mit einer Steckbrücke bestückt werden, da sonst keine fehlerfreie Kommunikation mit dem iSMT möglich ist.

Bild 10: Schaltbild der Schrittmotorsteuerung iSMT



Im Anschluss an die Bestückung kann das Modul entweder auf der Rückseite eines Schrittmotors mit den Abstandsrollchen und passenden Schrauben montiert werden oder mit Hilfe der beiliegenden Winkelbleche, wie in Bild 11 gezeigt, seitlich befestigt werden. Dem Bausatz liegt bereits eine kleine Auswahl an Montagematerial (Bild 12) bei, mit dem in vielen Fällen das iSMT-Modul befestigt werden kann. In Bild 13 sieht man deutlich die verschiedenen Montagelöcher in der Platine für unterschiedliche Motorgrößen.

Aufgrund der nahezu unendlich vielen Anwendungsgebiete und der unterschiedlichen Motoren muss von Fall zu Fall weiteres Montagezubehör eingesetzt werden.

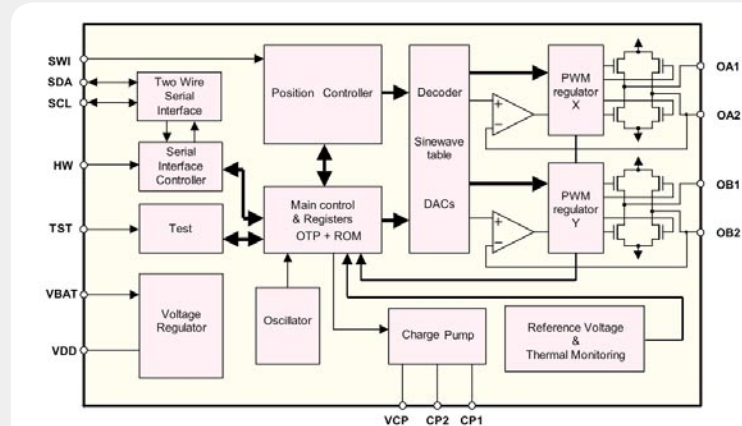
Zu beachten ist bei der Montage, dass Unterseite und Oberseite der Platine ausreichend Abstand (mindestens 2 mm) zum Motor und zur Umgebung einhalten. Um ausreichend ESD-Schutz zu gewährleisten, ist das Modul zudem (eventuell zusammen mit dem Motor) in ein Gehäuse einzubauen oder vollständig gegen Berührung abzuschirmen.

Weiterhin müssen alle abgehenden Leitungen (Spannungsversorgung, Motoranschlusskabel) kürzer als 3 m bleiben. Um Störungen in der Daten-

Schrittmotor-Treiber TMC223 (Trinamic)/ AMIS-30624 (ON Semiconductor)

Der Schrittmotor-Treiber integriert eine Kombination aus Motion-Controller, Treiber und zwei FET-Vollbrücken für einen bipolaren Schrittmotor in einem einzigen IC.

Der Treiber ermöglicht einen Mikroschritt-Betrieb mit bis zu 16 Mikroschritten je Vollschritt (4-Bit-Micro-Stepping) von Motoren mit einem Wicklungsstrom von 0,8 A (Iss) bzw. 0,57 A (Irms) bei einer zulässigen Motorspannung von 9 bis 29 V. Der Vollschritt-Betrieb ist bis zu einer Frequenz von 1 kHz möglich. Im Mikroschritt-Betrieb ist eine sehr exakte und leise Motorpositionierung möglich. Die PWM-Steuerung erfolgt mit einer festen Frequenz mit automatischer Ausführung von schnellem oder langsamem Ausschwingen. Ein integrierter RAM (OTP) speichert Motorparameter und Konfigurationen. Er führt nach der Konfiguration und Initialisierung alle zeitkritischen Operationen, basierend auf Zielposition und Beschleunigungsparametern (Beschleunigung, Verzögerung, Sanftanlauf), autonom aus. Die Kommunikation mit einem steuernden Controller (Host) erfolgt über ein serielles Zweidraht-Interface (TWI/I²C). Hierüber werden nicht nur Steuerbefehle, sondern auch Diagnose- und Statusmeldungen zum Host übermittelt.



Blockschaltbild des TMC223 von Trinamic.

Quelle: TRINAMIC Motion Control GmbH & Co. KG

Auf dem Chip sind zusätzlich Schutzschaltungen zur Erkennung von Übertemperatur, Überstrom, Kurzschluss, offenem Ausgang und Unterspannung integriert, so dass eine hohe Betriebssicherheit gewährleistet werden kann.

Der Controller verfügt über einen internen 16-Bit-Positionsähler, die Drehgeschwindigkeit und die Beschleunigung des Schrittmotors sind einstellbar und die Änderung der Zielposition ist während des aktiven Betriebs möglich.

Für den Anschluss eines externen Referenzschalters (z. B. Endposition) ist ein Steuereingang vorhanden.

Eine sensorlose, konfigurierbare Stall-Detection-Funktion erkennt eine Motor-Überlast, wie sie z. B. an einem Endanschlag auftritt, automatisch und hält den Motor an. So kann man zusätzliche Endschalter im mechanischen System einsparen.

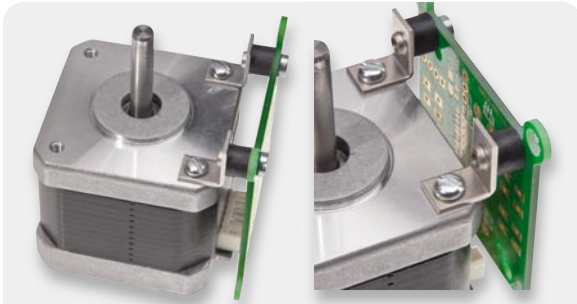


Bild 11: Die iSMT-Steuerung wird möglichst nahe am Motor montiert, wie hier gezeigt.



Bild 12: Das mitgelieferte Montagematerial erlaubt bereits zahlreiche Montagevarianten.

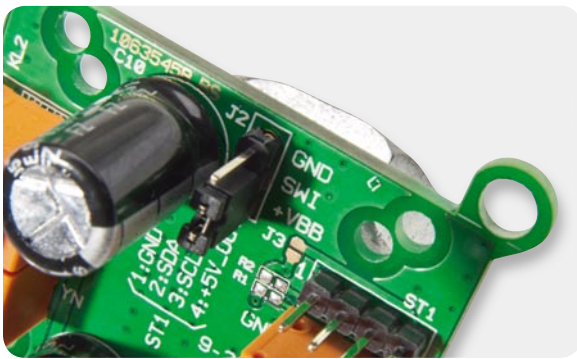


Bild 13: In der Platine sind Montagelöcher für die unterschiedlichen Motorgrößen vorhanden.

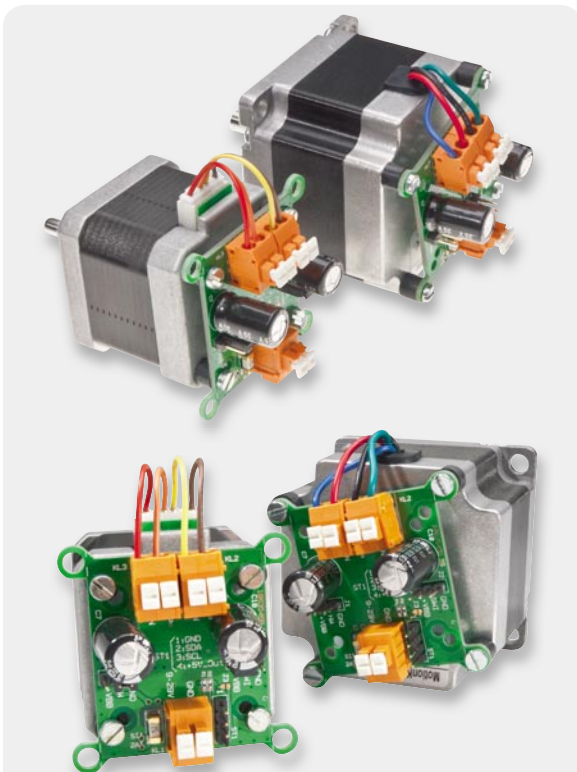


Bild 14: Zwei unterschiedliche Motorvarianten mit rückseitig montierten Steuerungsplatinen

Stückliste

Widerstände:

1 kΩ/SMD/1206 R3, R4

Kondensatoren:

4,7 nF/SMD/0805 C2, C6
 100 nF/SMD/0805 C1, C4, C8, C11
 220 nF/SMD/0805 C5, C9
 1 µF/20 V/Tantal/SMD C3
 100 µF/35 V C7, C10

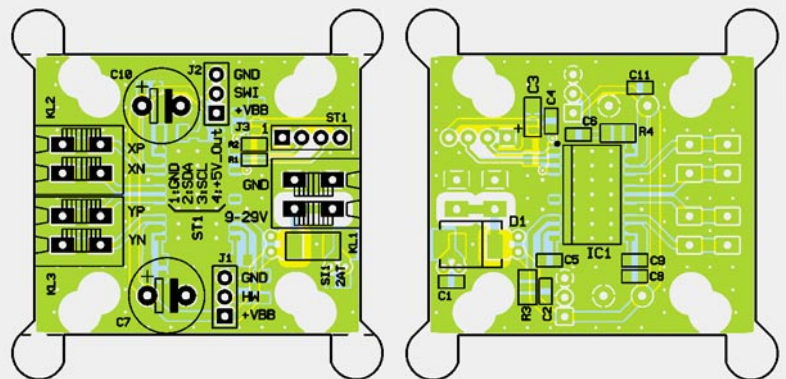
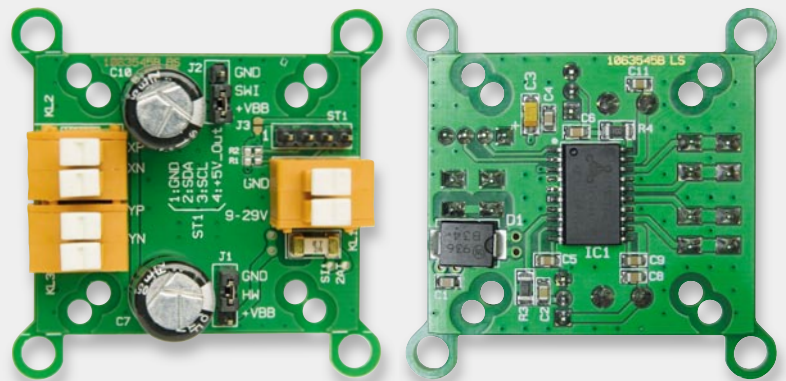
Halbleiter:

AMIS-30624 oder TMC223 IC1
 MBR340/SMD D1

Sonstiges:

Steckklemmleisten, 2-polig, 1,5 mm², Orange, print KL1, KL2, KL3
 Sicherung, 2 A, träge, SMD SI1
 Jumper ohne Griffflasche, geschlossene Ausführung J1, J2
 2 Stiftleisten, 1x 3-polig, gerade, print
 1 Stiftleiste, 1x 4-polig, gerade, print
 2 Zylinderkopfschrauben, M3 x 5 mm
 4 Zylinderkopfschrauben, M3 x 8 mm
 8 Fächerscheiben, M3
 4 Distanzrollen, M3 x 3 mm
 2 Befestigungswinkel, vernickelt

übertragung zu vermeiden, sollten die Leitungen der I²C/TWI -Schnittstelle je nach Taktrate nicht viel länger als 10 bis 50 cm sein. Bild 14 zeigt schließlich die montierte iSMT-Ansteuerung an den zwei unterschiedlichen, bereits vorgestellten MotionKing-Motoren. Hier erkennt man deutlich, dass je nach Motorausführung die Platine über unterschiedliche Löcher zu montieren ist.



Ansicht der bestückten Platine der iSMT-Schrittmotorsteuerung mit zugehörigem Bestückungsplan, links Oberseite, rechts Unterseite

Die Inbetriebnahme der iSMT in fünf Schritten

1. Den Schrittmotor am iSMT anschließen

Hat man einen Motor mit vier Anschlüssen, ist eine Motorwicklung an die Klemme KL 2 (XP, XN) und die andere an die Klemme KL 3 (YP, YN) anzuschließen. Die Polung ist dabei nur für die Drehrichtung wichtig. Wenn bei einem 4-poligen Motor nicht klar ist, welche Leitungen einer Wicklung zuzuordnen sind, kann man dies durch eine Widerstandsmessung mit einem Multimeter feststellen – die zusammengehörenden Anschlüsse haben einen eher niederohmigen Widerstandswert, bei den in Teil 1 vorgestellten neuen Schrittmotoren von MotionKing (14HY3402 und 23HS4412) sind das z. B. 6,5 bzw. 12 Ω .

Da die Steuerschaltung gegen alle denkbaren Fehler abgesichert ist, könnte man den korrekten Anschluss notfalls auch durch Ausprobieren herausfinden. Solch ein „unsauberes“ Vorgehen kann natürlich nur dann zum Erfolg führen, wenn für jeden Funktionstest genau die richtigen Befehlssequenzen gesendet werden. Da dies gerade bei den ersten Versuchen nicht immer sichergestellt werden kann, empfiehlt sich zum Einstieg auf jeden Fall die Messmethode.

Möchte man einen Motor mit 6 Anschlussdrähten am iSMT anschließen, muss man zunächst durch Widerstandsmessungen prüfen, welches die „äußeren“ Wicklungsabgriffe sind (siehe auch Bild 6, „ELVjournal“ 6/2010, Seite 49) und welche Leitungen die Mittelabgriffe bilden, die bei der bipolaren Ansteuerung nicht genutzt werden und daher isoliert werden müssen. Die richtigen Anschlüsse haben den jeweils größeren Widerstandswert, der aber natürlich nicht im Megaohmbereich liegen darf.

Schrittmotoren mit 8 Anschlüssen besitzen insgesamt 4 Wicklungen, von denen die jeweils zusammengehörigen entweder parallel oder in Serie geschaltet werden können, was zu unterschiedlichem Laufverhalten führt und am besten ausprobiert werden sollte. Bild 6 („ELVjournal“ 6/2010, Seite 49) zeigt den richtigen Anschluss der verschiedenen Motoren am iSMT.

2. Jumper J 1 und J 2 richtig stecken

Die zwei Positionen des Jumpers J 1 legen den Zustand des Bit 1 (HW) der I²C-Adresse fest. Wird HW mit GND verbunden, ist Bit 1 gleich 0, und wird HW mit +VBB verbunden, ist es gleich 1. Der HW-Anschluss an Jumper J 1 kann bei Bedarf auch zusammen mit der Busleitung über ein Steuerkabel geführt werden, wodurch bei Kabelbruch (HW-Leitung weder mit GND noch mit +VBB verbunden) der Motor automatisch die Sicherheitsposition (Secure Position) anfahren kann.

Statt eine Codierbrücke auf eine der beiden Positionen von Jumper J 2 zu setzen, kann an diesem 3-poligen Anschluss z. B. auch ein Motor-Positions-Schalter angeschlossen werden, dessen Zustand per Befehl abgefragt werden kann. Eine Zustandsänderung führt hier aber nicht zu einer weiteren Aktion. Der SWI-Pin ist also quasi nur ein per Software abfragbarer Eingang. Die Verbindung des SWI-Pins mit GND oder mit +VBB führt zum Zustand „Close“ (ESW = 1), während eine offene Verbindung zu „Open“ (ESW = 0) führt. Der Jumper J 2 kann also, wenn diese Funktion nicht benötigt wird, ignoriert werden.

3. Datenleitung (I²C/TWI) anschließen

Als Nächstes ist die Schrittmotor-Ansteuerung iSMT mit einem Mikrocontrollersystem oder mit dem USB-I2C-Interface von ELV zu verbinden. Dazu sind die Masse- (GND), die Daten- (SDA) und die Taktleitung (SCL) der Stiftheiste ST 1 mit dem Master-System zu verbinden. Bild 15 zeigt beispielhaft den Anschluss am USB-I2C-Interface.

Die +5-V-Leitung von ST 1 wird dabei nicht verbunden, da dies eine Ausgangsversorgungsleitung ist, die hier nicht benötigt wird (J 3 sorgt auf der iSMT-Platine zur Sicherheit dafür, dass diese Leitung spannungsfrei bleibt).

Die für eine TWI/I²C-Schnittstelle notwendigen Pull-up-Widerstände an SDA und SCL sollten sich entweder im Master-System befinden, wie es beim USB-I2C-Interface bereits der Fall ist, oder können alternativ auf der iSMT-Platine (R 1, R 2) nachträglich mit 10-k Ω -Widerständen (SMD-

Bauform: 0805) bestückt werden. Dabei ist darauf zu achten, dass die Pull-up-Widerstände nur gegen +5 V geschaltet werden dürfen, damit der Spannungspegel innerhalb des zulässigen Bereichs bleibt. Für höhere oder geringere Buspegel (z. B. +3,3 V) sind SDA und SCL über Pegel-Wandler anzuschließen – siehe dazu die FAQ „TMC211_TMC222_FAQ.pdf“ unter [6].

4. Spannungsversorgung anschließen

Im vorletzten Schritt der Inbetriebnahme muss das iSMT-Modul nur noch mit einer Gleichspannung zwischen 9 und 29 V versorgt werden, die über die Klemme KL 1 zuzuführen ist. Das verwendete Netzteil sollte mindestens den zweifachen, über die Register „IHold“ und „IRun“ eingestellten Phasenstrom bereitstellen können oder sicherheitshalber mindestens 2 A.

5. Befehle über I²C/TWI-Schnittstelle übertragen

Nach diesen Vorbereitungen kann die Ansteuerung des angeschlossenen Schrittmotors beginnen. Dies kann entweder über eine Mikrocontrollersteuerung oder viel einfacher direkt vom PC über das USB-I2C-Interface erfolgen. In beiden Fällen sollte man spätestens an dieser Stelle das Datenblatt des TMC223 oder des AMIS-30624 zur Hand nehmen und die Funktion der Steuerbefehle bzw. der Diagnosewerte nachschlagen. Am besten nutzt man hier, wie erwähnt, sogar beide Datenblätter, da diese sich hervorragend ergänzen und trotz gleicher Funktionsweise der Treiberbausteine sehr unterschiedlich geschrieben sind.

Den schnellsten Erfolg erzielt man sicherlich mit Hilfe des in Bild 8 („ELVjournal“ 6/2010, Seite 50) gezeigten und bereits besprochenen „iSMT-Test-Tools“, was folgendermaßen geschieht:

5.1 Nach der Installation und dem Start des Programms ist zuerst der richtige COM-Port auszuwählen und über „Setup“ die richtige Baudrate auszuwählen, wenn diese auf dem USB-I2C-Interface von der Werkseinstellung (115.200 bit/s) auf einen anderen Wert umgestellt worden ist.

5.2 Über die Schaltfläche „Connect“ wird zuerst der COM-Port geöffnet und dann die Konfiguration des USB-I2C-Interface abgefragt. Führt das Interface noch ein Makro aus, wird dieses nun beendet, da es sonst keine anderen Befehle ausführen könnte. Sobald die Verbindung hergestellt ist, ändert sich die Beschriftung der „Connect“-Schaltfläche in „Disconnect“. Nun ist rechts die I²C/TWI-Adresse einzustellen, die im Werkszustand je nach Stellung des Jumpers J 1 entweder 192 (GND) oder 194 (+VBB) ist. Das Read/Write-Bit 0 der Adresse ist hier stets 0, was sie als Schreib-Adresse kennzeichnet. Bei Lesezugriffen verwendet das iSMT-Tool automatisch die Lese-Adresse, bei der Bit 0 immer 1 ist.

5.3 Nun kann ein erster Test durchgeführt werden, indem auf der iSMT-Registerkarte zuerst die Unterregisterkarte „Get“ und dort die Schaltfläche „GetFull-Status1“ gedrückt wird. Sofort sollten sich die Felder daneben mit Daten füllen und die Anzeigen über den Zustand ihrer Status-Bits informieren. Als Nächstes sollte man über „GetFullStatus2“ weitere Informationen aus dem RAM des iSMT auslesen und im Datenblatt nachschlagen, was diese zu bedeuten haben.

Einige Statusparameter werden durch das Auslesen bereits zurückgesetzt, so dass bestimmte, durch das Einschalten erfolgte Warnungen (VddReset, StepLoss, UV2, ELDef, OVC1, OVC2) beim zweiten Auslesen verschwinden sollten. Die vollständige bidirektionale Kommunikation mit dem USB-I2C-Interface wird im Protokollbereich unten aufgezeichnet, woraus sich Funktion und Zusammensetzung der Befehle gut nachvollziehen lassen.

5.4 Als Nächstes sollte man auf der Unterregisterkarte „Set“ geeignete Motorparameter einstellen. Nach dem Einschalten stehen die meisten Werte auf 0, was dem kleinsten Motorstrom, dem Halbschrittbetrieb und der geringsten Drehgeschwindigkeit und -beschleunigung entspricht. Vielen Motoren genügt der geringe Strom von nur 59 mA aber nicht zum Anlaufen, weshalb der IRun-Wert höher gesetzt werden sollte, z. B. auf 7, was 200 mA entspricht. Übertragen werden die eingestellten Werte mit „SetMotorParam“.

5.5 Nun kann man den Motor entweder mit dem Befehl „SetPosition“ auf die nebenstehend anzugebende Position bewegen oder auf die Registerkarte „Run/Stop“ wechseln und mit dem Befehl „RunVelocity“ die Motorachse endlos drehen lassen. Während der Motor dreht, können die Motorparameter geändert und erneut übertragen werden. So lassen sich schnell die unterschiedlichen Einstellungen austesten.

5.6 Mit „SoftStop“ lässt sich der Motor je nach eingestellter Beschleunigung langsam stoppen, während „HardStop“ ihn sofort zum Stehen bringt.

Generell zu beachten ist, dass nach dem Auftreten von Fehlern und auf eine Reihe von Befehlen folgend wieder mit „GetStatus1“ und „SetMotorParam“ begonnen werden sollte, da der Motor sonst nicht anläuft!

Wie die Kommunikation mit dem iSMT über einen I²C-Bus funktioniert und wie sich die Befehlssequenzen zusammensetzen, kann hier nicht erklärt werden, da dies den Umfang des Artikels vervielfachen würde. Ausführliche, gut verständliche Informationen zu diesen Themen finden sich z. B. in der Dokumentation zum USB-I2C-Interface und im Datenblatt des AMIS-30624 (siehe Kapitel 15.0 in [5]). Weiterhin findet sich als Download auf der iSMT-Produktseite die Befehlssequenzliste „iSMT_Befehlsliste.hts“, die mit dem Terminalprogramm „HTerm“ geladen und zur Kommunikation mit dem iSMT über das USB-I2C-Interface genutzt werden kann.

Falls kein USB-I2C-Interface zur Verfügung steht und die Kommunikation über einen Mikrocontroller realisiert werden soll, gibt es auf der Trinamic-Webseite [6] die bereits erwähnte AVR-C-Library „TMC222_Lib_AVR.zip“, mit deren Hilfe die Einbindung des iSMT

schnell und relativ einfach vonstatten geht. C-Einsteiger finden unter [9] umfangreiche Grundlagen und viele hilfreiche Beispiele zur Programmierung in C mit dem WINAVR-Compiler. Auf „mikrocontroller.net“ finden sich zudem sehr nützliche Artikel zu „I²C“, „AVR TWI“ und einige weiterführende Forenbeiträge zum TMC222, der mit dem TMC223 bis auf die fehlende Blockade-Erkennung weitestgehend identisch ist. **ELV**



Weitere Infos:

- [1] Ausführliche Infos und Forum zur Realisierung von Robotersteuerungen und zu Schrittmotoren: www.roboternetz.de
- [2] Infos und Forum zum Bau von CNC-Maschinen: 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
- [6] Datenblatt, AVR-C-Library, TMC223-FAQ und weitere Infos zum Trinamic TMC223: www.trinamic.com/tmc/render.php?sess_pid=394
Deutschsprachiges Forum zum Trinamic TMC223: www.trinamic.com/ttdg/viewforum.php?f=16
- [7] iSMT-Produktseite: www.elv.de, bitte die Bestellnummer 92720 im Suchfeld eingeben
- [8] Das Terminalprogramm HTerm: www.der-hammer.info/terminal
- [9] Grundlagen, Tipps, ein Forum und viele Beispiele zu AVR-Programmierung: www.mikrocontroller.net/articles/AVR-GCC-Tutorial

