

USB-I²C Stick



Der USB-I²C Stick ist eine Steuerung mit integriertem Mikrocontroller. Der USB Anschluss eines Computers wird als virtueller COM-Port verwendet, um den USB-I²C Stick aus einer Software anzusprechen. Als Ausgang steht ein I²C Signal zur Ansteuerung der TB-ELECTRONICS Bausteine. Der Anwender oder Programmierer muss sich nicht um die I²C spezifische Signalerzeugung kümmern.

Der USB-I²C Stick verwendet fest I²C Adressen. Diese sind hier dokumentiert. Ausserdem wird ein komplexer Klartext Befehlssatz verwendet, der ausschliesslich ASCII Zeichen verwendet. Damit kann ein so betriebenes System auch mit einfachen Terminalprogrammen auskommen, um den I²C Bus anzusprechen.

Grundsätzlich gilt:

- Der USB-I2C Stick verwendet einen Befehlsbuffer in der Grösse von 1000 Zeichen.
- Als Befehlstrennzeichen (delimiter) werden die Zeichen <;> oder char[13] = 0x0d verwendet. In dieser Beschreibung wird als Delimiter ausschliesslich das <;>-Zeichen verwendet.
- ein unbekannter Befehl wird mit 'SYNTAXERROR' quittiert.
- Alle Befehle unterscheiden Gross- und Kleinschreibung.
- Alle Quittierungen bzw. Ausgaben enden mit dem <;>-Zeichen.
- Wird in der unten stehenden Befehlsbeschreibung als Antwort "z.B. " aufgeführt, so ist die Antwort abhängig vom Parameter "confirm on".
- Der virtuelle COM-Port muss folgendermassen eingestellt sein:
 - 38400 baud
 - 8 Datenbits
 - 1 Stopbit
 - kein Hardwarehandshake, kein Xon/Xoff
- Wenn nicht explizit anders erwähnt, dann handelt es sich bei den I2C Busteilnehmern um TMC222 Schrittmotorcontroller bzw. SM32 Baugruppen von TB-ELECTRONICS. Zu Fragen und Betrieb der TMC222 muss die technische Dokumentation von Trinamic verwendet werden.
- Geänderte Parameter werden NICHT im USB-I2C Stick gespeichert.

Stromaufnahme aus dem USB-Port:

< 80mA

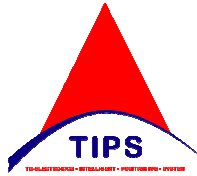
Treiber-Details:

Silabs Treiber für CP210x virtuellen ComPort

Anschluss:

3-polig über 5-polige Buchsenleiste

- SCL = gelb
- SDA = orange
- GND = schwarz

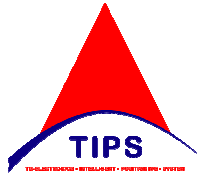


USB-I²C Stick

Befehle in alphabetischer Reihenfolge

Systembefehle

Befehl	Bedeutung
buf;	Zeigt den aktuellen Stand des Empfangs- buffers. Antwort: BUFFER=15;
confirm on;	Jeder eingegebene Befehl wird mit der Antwort quittiert (default). Antwort: CONFIRM=ON;
confirm off;	Die Quittierung von Befehlen wird unterdrückt. Antwort: CONFIRM=OFF;
echo on;	Jedes eingegebene Zeichen wird quittiert. Antwort: ECHO=ON;
echo off;	Ein eingegebens Zeichen wird nicht quittiert (default). Antwort: ECHO=FF;
new;	Rücksetzen des Eingabebuffers auf 0. Antwort: BUFFER=0;
nl;	Neue Zeile (new line). <cr> <lf>
delay vvvvv;	Verzögert den Befehlsablauf. vvvvv=0 ... 65535. v=1 entspricht 1 Millisekunde. Antwort: delay...ready; „ready“ wird nach Ablauf der Delayzeit angezeigt.



USB-I²C Stick

whois;

Zeigt alle Schrittmotor-Busteilnehmer an und initialisiert die Schrittmotoren mit Default Werten. Es wird für alle Motoren s=3 eingestellt. Das entspricht der kleinsten Schrittauflösung der TMC222.

Ein korrekt angeschlossener Motor wird mit seiner Adresse angezeigt. Ansonsten erfolgt.

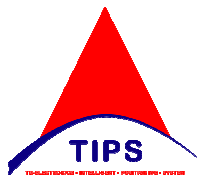
Antwort z.B. wenn nur MOT3 und MOT7 installiert sind:

MOT=3=0xCC;
MOT=7=0xEC;

version;

Zeigt die aktuell installierte Firmware

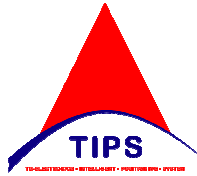
Antwort: V1.00B01;



USB-I²C Stick

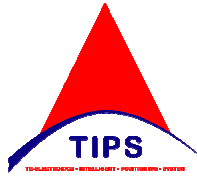
Motorsteuerbefehle - nicht in alfabetischer, sondern in funktionell logischer Reihenfolge.

Befehl	Bedeutung	Antwort
mot v;	<p>Auswahl des anzusteuernenden Schrittmotors. Motorennummer v=0...13. Wenn v=0, dann wird Adresse 0xC0 als I²C device address verwendet. Das entspricht einem nicht programmierten TMC222. Das Leerzeichen zwischen 'MOT' und dem Wert muss nicht vorhanden sein. Alle auf den MOT-Befehl folgende Befehle beziehen sich auf die Motorennummer.</p> <p>Beispiel: mot7; Antwort: MOT=7=0xEC;</p>	z.B. 'MOT=
maccel vv;	<p>Setzen der Beschleunigung des aktuell angewählten Motors. vv=1...15</p> <p>Kleine Werte = kleine Beschleunigung, grosse Werte = grosse Beschleunigung. Default: vv=2.</p> <p>Beispiel: maccel3; Antwort: Accel=3;</p>	
mihold vv;	<p>Setzen des Haltestroms des aktuell angewählten Motors. vv=0...15</p> <p>Kleine Werte = kleiner Haltestrom, grosse Werte = grosser Haltestrom. Default: vv=2.</p> <p>Beispiel: mihold1; Antwort: lhold=1;</p>	



USB-I²C Stick

- mirun vv; Setzen des Fahrstroms des aktuell angewählten Motors. vv=0...15
- Kleine Werte = kleiner Fahrstrom,
grosse Werte = grosser Fahrstrom.
Default: vv=10.
- Beispiel: mirun13;
Antwort: Irun=13;
- mvmin vv; Setzen der Minimalgeschwindigkeit des aktuell angewählten Motors. vv=0...15
- Kleine Werte = kleine Geschwindigkeit,
grosse Werte = grosse Geschwindigkeit.
Default: vv=2.
- Beispiel: mvmin1;
Antwort: Vmin=1;
- mvmax vv; Setzen der Maximalgeschwindigkeit des aktuell angewählten Motors. vv=0...15
- Kleine Werte = kleine Geschwindigkeit,
grosse Werte = grosse Geschwindigkeit.
Default: vv=10.
- Beispiel: mvmax12;
Antwort: Vmax=12;
- mshaft v; Setzen der Richtung für positive Positionswerte. v=0...1.
- Default: v=0.
- Beispiel: mshaft1;
Antwort: Shaft=1;



USB-I²C Stick

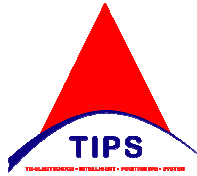
- mgo vvvvv;
Setzen der Zielposition des aktuell angewählten Schrittmotors und Start der Fahrbewegung.
-32000 < vvvvv < +32000 Inkremente.

Beispiel: mgo12000; fährt den Motor auf Position 12000 Schritte.
- mpos;
Abfrage der Position des aktuell angewählten Motors.

Beispiel: mpos12000;
Antwort: MOT7=+12000;
- msearch v;
Suche der Referenzposition des aktuell angewählten Motors.
v=0...1.
v=0 fährt solange bis der angeschlossene Endschalter schliesst.
v=1 fährt solange bis der angeschlossene Endschalter öffnet.
Wenn die Zeitüberwachung abgelaufen ist, wird der Vorgang beendet.
- mstart v;
Betrieb des aktuell angewählten Motors im Dauerlauf - keine Positionierung.

v=0...2.
v=0 stoppt den Dauerlauf.
v=1 startet in Standard-Drehrichtung.
v=2 startet in Umkehr-Drehrichtung.

Beispiel: mstart1;
Antwort: Start=1; Motor dreht solange, bis der Befehl mstart0; ausgelöst wird.



USB-I²C Stick

Beispiel für einen Befehlsablauf:

```
new;mot3;mspeed3;mgo10000;mot2;mgo3000;delay1500;mgo0;mot3;mgo0;
```

- rücksetzen des Befehlsbuffers
- Motor 3 adressieren
- Geschwindigkeit MOT3 auf 3 setzen
- Motor 3 auf 10000 positionieren
- Motor 2 adressieren
- Motor 2 auf -3000 positionieren
- 1500 Millisekunden warten
- Motor 2 auf 0 positionieren
- Motor 3 adressieren
- Motor 3 auf 0 positionieren

Design and Copyright by

TB-ELECTRONICS GmbH

Bahnhofstrasse 3
CH-9443 Widnau



Tel.: 071 722'52'55
Fax: 071 722'52'05

Änderungen ohne weitere Vorankündigungen vorbehalten. Dieses Dokument und Auszüge daraus unterliegen dem internationalen Copyright. Kopien dürfen nur für den eigenen Bedarf angefertigt werden.