4.2 Beschreibung der Datenschnittstelle

a.) Centronics (nur für SMC800 und SMC1500 ohne Zusatzplatine)

Bezeichnung	Funktion
STROBE	Datenübernahme (low aktiv)
DATA1/DATA2	Interne Stromquelle der Motorendstufe (Phase A)

DATA 1	DATA 2	
1	1	kein Strom
0	1	minimaler Strom (20% des max. Stroms)
1	0	mittlerer Strom (60% des max. Stroms)
0	0	maximaler Strom

DATA 3	Richtung des Phasenstroms (Phase A)
DATA4/DATA5	Interne Stromquelle der Motorendstufe (Phase B) siehe DATA1/DATA2
DATA 6	Richtung des Phasenstroms (Phase B)
DATA7/DATA8	Motorendstufe selektieren

DATA 7	DATA 8	
0	0	Motorendstufe X
1	0	Motorendstufe Y
0	1	Motorendstufe Z

5. Technische Daten

	SMC800	SMC1500
Versorgungsspannung:	14V ... 28V DC	24V ... 28V DC
Stromaufnahme:	max. 5A	max. 9,5A
Phasenstrom:	5mA ... 800 mA	5mA ... 1500mA
Arbeitskontakt:		24V/8A AC (nur mit Zusatzpl.)
Referenzschalter:	mech. Schließer	mech. Schließer
Eingangspegel:	TTL kompatibel	TTL kompatibel (Centronics) TTL und CMOS kompatibel (mit Zusatzplatine)
Eingangsfrequenz:	max.. 5kHz	max. 5kHz

Stepper Motor Card
SMC800 / SMC1500

Technical

Documentation

Table of contents

1. Introduction

The stepper motor card SMC800/SMC1500 is for controlling of bipolar stepper motors in a range of small to middle performance.

The stepper motor card can be controlled by different computers via parallel port (Centronics). With the additional PCB SMC1500Z other controlling devices can be used.

A variable external power supply and continuous adjustable phase currents ensures simple adaptation to different motor types.
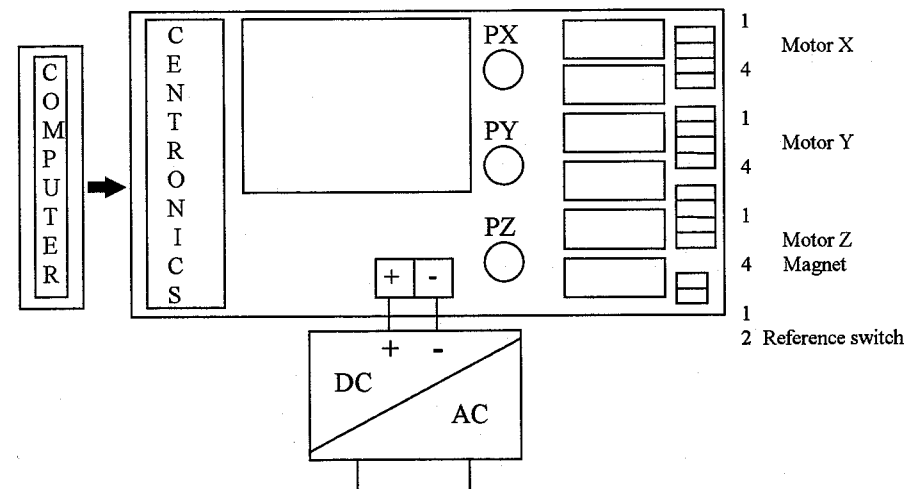
For more information about the theory of stepper motors please refer to the following literature:

> Schrittmotor-Antriebe
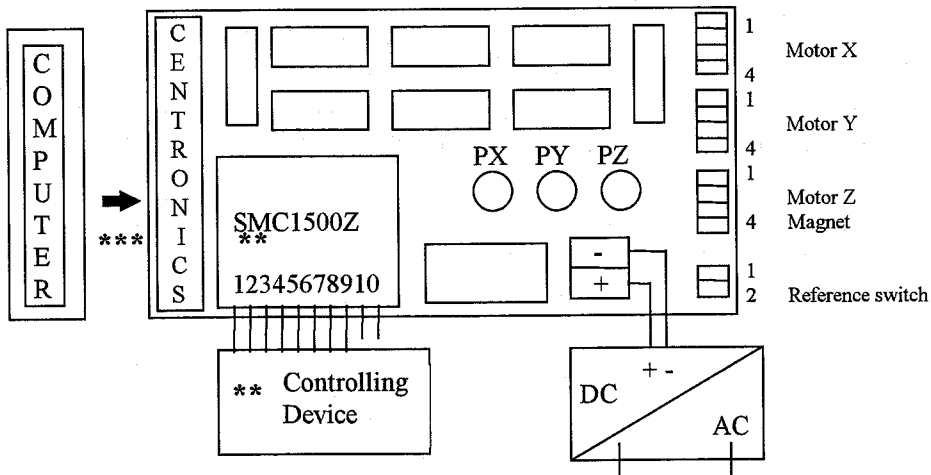> Franzis-Verlag GmbH
> Author: Prautzsch

2. Hardware

2.1 Overview

a.) SMC800

b.) SMC1500

```
COMPUTER  →  CENTRONICS    SMC1500Z **    PX PY PZ
                           12345678910
```

***

Motor X

1
4
1
Motor Y

4
1
Motor Z

4
Magnet
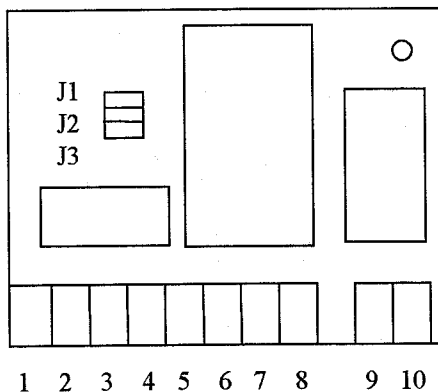
1
2
Reference switch

** Controlling Device

DC + -
AC

**   Only for SMC1500 with additional PCB SMC1500Z
        (Upgrade possible)

***   Only one data interface can be used at the same time (Centronics or
        connector at additional PCB SMC1500Z)

c.) Additional PCB SMC1500Z

```
J1
J2
J3

1 2 3 4 5 6 7 8   9 10
```

Connector description:
1: Clock signal X-Motor
2: Direction signal X-Motor
3: Clock signal Y-Motor
4: Direction signal Y-Motor
5: Clock signal Z-Motor
6: Direction signal Z-Motor
7: Relay-front contact on/off (coil)
8: Relay-Ground (coil)
9: Relay-front contact connect 1
10: Relay-front contact connect 2

The additional PCB SMC1500Z (only possible to use with SMC1500) is for controlling the stepper motor card by clock and direction signal. The PCB includes a PIC16C57 microcontroller and provides the following features:

- Full or half step mode
- Hold current at standstill motor

The corresponding jumper settings for J1, J2 and J3 are in the following tables.

	J1
Half step	ON
Fullstep	OFF

Hold current	J2	J3
0%	OFF	OFF
20%	ON	OFF
60%	OFF	ON

When the motor stops the phase current will be reduced automatically to the adjusted value.

The input signals are TTL and CMOS compatible and working with a maximum frequency of 5kHz. The trigger for the clock and direction inputs is the rising edge.

By changing the signal at the direction input ($0 \rightarrow 1$ or $1 \rightarrow 0$) the turning direction of the stepper motor will change. The relay will be switched on with positive voltage level (3-30V) and switched off with zero level (0-0,8V).

The new jumper settings after changing will get active by a power up or a reset (interrupt the power supply).

2.2 Power Supply

For running the stepper motor card an external DC power supply is necessary. The residual ripple of the DC power supply should not be higher than 2 Volt.

Attention! High voltage (U > 30V) can destroy the stepper motor card.

2.3 Motor Connection

A 2-phase stepper motor will be connected to one of the motor outputs as shown in the following picture. One phase is connected to pins 1-2 and the other phase is connected to pins 3-4.
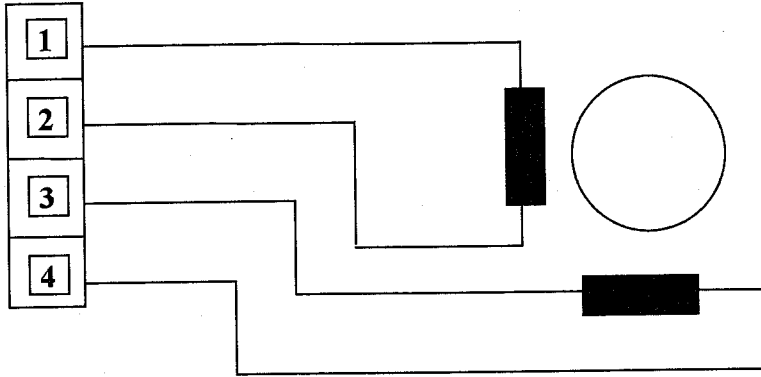
When a 4-phase stepper motor should be connected to one of the motor outputs two phases must be switched in series as shown in the following picture.
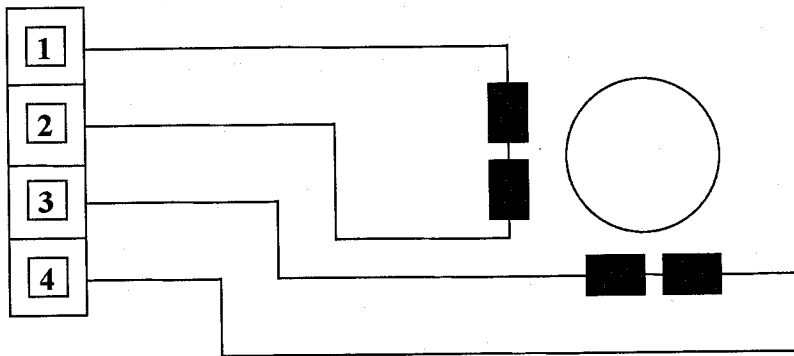
Turning direction of the motor
The turning direction of a stepper motor can be changed by an exchange of the wires of one phase.

Example: 2-phase stepper motor: Phase 1: Pins 1-2
                                         Phase 2: Pins 3-4

By exchange of the wires 1-2 the turning direction will change.
The same effect will happen at wires 3-4.

2.4 Phase Current Adjustment

The phase currents can be adjusted continuos in a range from 5 mA to 800 mA (SMC800) and from 5 mA to 1500 mA (SMC1500).

The total current is assembled out of the single phase currents as following:

$$I_M = I_{Ph1} + I_{Ph2}$$

Adjustment of phase currents:

- Connect the motors to the stepper motor card.
- Connect an external power supply to the stepper motor card.
- The phase current can be derived from the reference voltage.

The reference voltage can be measured between the middle connector of the corresponding potentiometer (PX/PY/PZ) and ground. The potentiometer must be adjusted until the wanted phase current is reached according to the following formula:

$$I_{Ph} = V_{Ref} * 0,084 \text{ [A]} \qquad (\text{SMC800})$$
$$I_{Ph} = V_{Ref} * 0,123 \text{ [A]} \qquad (\text{SMC1500})$$

I_{Ph} : Phase current
V_{Ref} : Reference voltage

Attention: Users of the Stepper Motor Card **SMC800** should pay attention to use ventilation for the motor drivers at phase currents higher than 500mA.

2.5   Reference Switch

The stepper motor card has a connector where a reference switch can be plugged. The included software supports switches with make contact.

More than one reference switches (e.g. one for each axis) can be used and have to be plugged in parallel. Refer to *Chapter 3.2.*

2.6 Magnet for Plotter Mode

In plotter mode the Z-axis will be used to control a magnet or a relay instead of a stepper motor. In this mode only one phase is used (pin 1-2).

The voltage for the magnet depends on the stepper motor voltage:

Example:　　Power supply for stepper motor card: 24V DC

　　　　━━━━➤ Power supply for magnet: 24V DC

The maximum current is 800 mA (SMC800) and 1500 mA (SMC1500). If a more powerful magnet wants to be used, a relay has to be connected in between. For the relay the same conditions has to be observed.

2.7　Controlling of the Stepper Motor Card

2.7.1　Connection to a PC
The stepper motor card can be connected to the parallel port (Centronics) of a PC. For this a standard printer cable is necessary.

2.7.2　Connection to other Controlling Devices
Only SMC1500 with additional PCB !
The stepper motor card SMC1500 equip with additional PCB SMC1500Z can be controlled from a PC or other controlling devices (e.g. SPS, Frequency generator ...). The controlling device has to give clock and direction signal to the stepper motor card.

3. Software

The driver software delivered with the stepper motor card gives the user the possibilities to run the stepper motor card with a special controlling software or to implement the stepper motor card specific source code directly into the own software.

3.1 Controlling Software
On the enclosed disk there are the following directories with the corresponding driver programs and sources:

3.1.1 SMC_DRV1:

This directory includes the driver for SMC800 and SMC1500 (without additional PCB SMC1500Z). The controlling of the stepper motor card depends on the speed of the PC. The optimum output frequency value has to be ascertained by trying.

For direct controlling of the stepper motor card SMC800 and SMC1500 the program SMC800.EXE is included. The program generates clock, direction and current information and sends it to the stepper motor card which converts it into corresponding phase currents to move the attached motors.

The output frequency of the information signals depends on the speed of the PC (e.g. as faster the PC the faster the output of clock, direction and current information). Therefore it is necessary to adapted the software to the PC. (Menu **System parameters**)

Following the system parameters are described in detail.

Z-axis (Z-Achse):
The Z-axis allows two kinds of controlling possibilities:
- Controlling in plotter mode: Instead of a stepper motor a magnet will be controlled. The value -32000 or 32000 has to be put in.
- Controlling in 3-axis mode: The Z-axis will be controlled like the X- and the Y-axis. A value in the range of –31999 to 31999 (= number of half steps) has to be put in.

Refdistance X (Refabstand X):
After reaching the reference switch the stepper motor will move the number of half steps put in at *Refdistance X* away from the reference switch. The value can be in the range of –31999 to 31999.

Refdistance Y (Refabstand Y):
Refer to Refdistance X.

Refdistance Z (Refabstand Z):
In 3-axis mode refer to *Refdistance X*. In plotter mode the value is not used.

Start-Ramp (Start-Rampe):
The value *Start-Ramp* represents the start up frequency of the stepper motor. It can be in the range of 1 to 32000. The value is reciprocal to the frequency (that means high value is low frequency of stepper motor). The value depends on the speed of the PC too.

End-Ramp (Ende-Rampe):
The value *End-Ramp* represents the working frequency of the stepper motor. It can be in the range of 1 to 10000. The value is reciprocal to the frequency (that means high value is low frequency of stepper motor). The value depends on the speed of the PC too.

Slope (Steilheit):
The *slope* value is the relative acceleration of the stepper motor. As higher the value the higher the acceleration and the faster the working frequency will be reached. The value can be in a range of 1 to 20.

Manual speed (Manuelle Geschwindigkeit):
This value represents the speed while driving the stepper motor by the keys. The value can be in a range of 1 to 30.

Interface (Schnittstelle):
This value selects the parallel port (Centronics). The following values can be used:

1	(LPT1)
2	(LPT2)

The system parameters can be saved by pressing **F9**.
By pressing **ESC** this menu can be left.

After setting the system parameters, positioning instructions can be sent to the stepper motor card via the menu **Positioning**.

The following program functions are available:

Reference (Referenz):
The selected stepper motor moves with slow speed in the negative direction. After reaching the reference switch the direction will be changed and the stepper motor moves in positive direction until the number of half steps put in at *Refdistance X, Y, Z* is reached. This operation continues with the Y-axis and the Z-axis in the same way.

In plotter mode (Z-axis used for magnet) while reference drive it can be selected whether the Z-axis is on current or not.

The reference positioning brings the system into a zero position depending on the position of the reference switches.

The reference positioning can be stopped by pressing **ESC**.

Manual (Manuell):
Positioning by using the keys (using the settings for manual speed).

Keys:		
	Cursor left / right :	X-Motor - Positioning
	Cursor up / down :	Y-Motor - Positioning
	Page up / down :	Z-Motor - Positioning or. relay on / off

Output file (Dateiausgabe):

This function can send vector files to the stepper motor card. This vector files can be programmed via menu **File (Datei)**. After selecting a vector file the programmed vectors will be send to the stepper motor card by using the system parameter settings. This operation can be stopped by pressing **ESC**.

The menu **File (Datei)** is for programming stepper motor card vectors.

Via an editor for every axis (X,Y,Z) the number of half steps can be programmed and saved.

The menu **File (Datei)** consists of the following functions:
- **New (Neu):**   Program a new vector file
- **Open (Öffnen):**   Edit an existing vector file
- **Exit (Beenden):**   Exit the program

After choosing the function **New (Neu)** or **Open (Öffnen)** an editor for setting the vectors can be used.

There are several input fields for every axis (X/Y/Z) to put in the vectors for the stepper motors. The input fields can be selected by using the keys (right/left/up/down) and (Page Up/ Page Down).

The input values are half steps. The values are directly send to the stepper motor card where the stepper motors run the number of half steps. The values can be in the range of –32768 to +32767. Negative values will cause movement into negative direction and positive values in positive direction. For the value 0 there is no movement of the corresponding axis.

Line by line the number of half steps for the stepper motors of the corresponding axis can be set. So, it will be set which motors are moving at the same time. The steps will be send line by line to the stepper motor card (linear interpolation).

The following table gives an example:

	X-Axis	Y-Axis	Z-Axis
Line 1	1000	0	0
Line 2	200	200	0
Line 3	0	0	500
Line 4	-300	0	-300
Line 5	0	-1000	0
Line 6	400	400	200

By pressing key **F2** the output of the vector file to the stepper motor card will start. The output will work in the following way:

At first line 1 will be sent to the stepper motor card. The motor at the X-axis will move 1000 half steps in positive direction and the stepper motors at Y- and Z-axis will not move. Then line 2 will be sent to the stepper motor card. The X- and the Y-axis stepper motors are moving 200 half steps in positive direction and the stepper motor at the Z-axis will not move. Afterwards line 3 will be next. The X- and the Y-axis stepper motors will not move and the stepper motor at the Z-axis will move 500 half steps in positive direction. In line 4 the X- and the Z-axis are moving 300 half steps in negative direction and the Y-axis will not move. Afterwards the Y-axis is moving 1000 half steps in negative direction while X- and Z-axis are not moving. Last in line 6 all three axis are moving. The X- and the Y-axis are moving 400 half steps and the Z-axis is moving 200 half steps in positive direction.

This example shows that the editor allows to produce complex movements which can be sent to the stepper motor card. Furthermore the axis can be moved separate or together (linear interpolation).

While output of the vectors the system parameters are used. So, in this example the 3-Axis mode is used (for Z-axis values in a range of −31999 to 31999 are used and to all three axis an stepper motor is connected).

The Z-axis also can be used in the plotter mode. In this case the value in the system parameters for the Z-axis must be set to −32000 or 32000 and a magnet must be plugged to the Z-axis. In the editor the values for the Z-axis will get a new meaning. Now only the positive or negative value for the Z-axis is important. It does not matter if the value is 1, 1000 or 10000 for a positive direction or −1, -1000 or −10000 for a negative direction.

The following table gives an example for the plotter mode:

	X-Axis	Y-Axis	Z-Axis
Line 1	1000	1000	0
Line 2	0	0	10
Line 3	400	0	0
Line 4	0	0	-1000

By pressing key **F2** the output of the vector file to the stepper motor card will start. The output will work in the following way:

At first line 1 will be sent to the stepper motor card. The motors at the X- and the Y-axis will move 1000 half steps in positive direction. At the Z-axis there will be no change. In line 2 the X- and Y-axis are not moving. Instead of moving the Z-axis (10 half steps) the output pins 1-2 of the Z-axis will be set on current (switching on the magnet). In line 3 only the X-axis will move 400 half steps in positive direction. The Y-axis will not move and the Z-axis will keep the current on. In line 4 the X- and the Y-axis will not move. The Z-axis will switch of f the current (switching off the magnet). The number of steps (-1000 half steps) is not important only the direction.

After putting in the values the defined vectors oft the X/Y/Z-axis can be saved with the key **F9**. For this the file name without extension has to be set. The extension is automatically set to (.SMC). Now the saved data can be sent to the stepper motor card with the menu **Output file (Dateiausgabe)**.

By pressing **F10** this menu can be left.

3.1.2 SMC_DRV2:

This directory includes the driver for SMC800 and SMC1500 (without additional PCB SMC1500Z). The controlling of the stepper motor card is independent of the speed of the PC. The output frequency value can be set in steps per second.

For direct controlling of the stepper motor card SMC800 and SMC1500 the program SMC800.EXE is included. The function principle is the same as described at SMC_DRV1.

Because of using steps per second there are new input fields in the system parameters. The input fields are described below:

Start frequency (Startfrequenz):
The value *Start frequency* is the start up frequency of the stepper motor. The value can be in the range of 1 to 5000 steps per second. The start frequency should be smaller than the working frequency.

Working frequency (Arbeitsfrequenz):
This value is the *Working frequency* of the stepper motor. The value can be in the range of 1 to 5000 steps per second. The working frequency should be higher than the start frequency.

Acceleration (Beschleunigung):
The value *Acceleration* defines the acceleration ramp. When sending a vector to the stepper motor card the stepper motor will start up with the start frequency. Then it will accelerate up to the working frequency. The value acceleration can be in the range of 1 to 5000 steps per second.

Break (Bremsen):
This value is for breaking the stepper motor from the working frequency down to zero. The value can be in the range of 1 to 5000 steps per second.

Reference frequency (Referenzfrequenz):
The value *Reference frequency* is the working frequency while reference positioning. The value can be in the range of 1 to 5000 steps per second. The reference positioning will be done without acceleration and breaking.

3.1.3 SMC_DV2A:

This directory includes the driver for SMC1500 with additional PCB SMC1500Z. The controlling of the stepper motor card is independent of the speed of the PC. The output frequency value can be set in steps per second.

For direct controlling of the stepper motor card SMC1500 the program SMC1500Z.EXE is included. The function principle is the same as described at SMC_DRV2.
This driver sends clock and direction signals via the parallel port (Centronics) of the PC to the stepper motor card. This signals only can be used by the additional PCB SMC1500Z.

3.2 Driver Software for C and Turbo Pascal

For the most applications it is necessary to include the stepper motor card controlling program parts directly into the own source code. For this case the source code of the drivers is available in the language C and Turbo Pascal.

The steps sent to the stepper motor card are half steps. So, it must paid attention to send also half steps to the motor driver routines.
The following stepper motor card driver files are included:

> SMCMOT.C (Motor driver in C)
> SMCMOT.PAS (Motor driver in Turbo Pascal)

There are example programs for the drivers (SMCMAIN.C, SMCMAIN.PAS) which demonstrate the controlling of the stepper motor card. The software can distinguish between plotter mode and 3-Axis mode.

Plotter mode:
For positioning of the X- and Y-axis the vectors must be sent to the stepper motor driver module SMCMOT. The values can be in the range of –31999 to 31999.
The phase 1-2 of the Z-axis will be switched on with value 32000 and switched of with –32000. So, a relay or magnet can be controlled. The value will be sent to the stepper motor driver module together with the X- and Y-axis values.

3-Axis mode:
For positioning of the X-, Y- and Z-axis the vectors must be sent to the stepper motor driver module SMCMOT. The values can be in the range of –31999 to 31999.

The following things must be observed by using the motor drivers:

a.) Global declaration

SMC_DRV1:

Variable	Data type in C	Data type in TP	Init value
MPTRX	unsigned char	byte	0
MPTRY	unsigned char	byte	0
MPTRZ	unsigned char	byte	0
RampGen	unsigned char	boolean	false
_LPT	unsigned int	word	378h = LPT1
			278h = LPT2

If a PC with MDA-card is used the following addresses of the parallel port (Centronics) have to be used: 3BCh = LPT1
378h = LPT2

At program start up the variables MPTRX, MPTRY, MPTRZ and RampGen must be initialised with the value 0.

For the output of the signals to the stepper motor card the address of the parallel port (Centronics) has to be assigned to the variable _LPT at the program start up. For most of the PCs the values for LPT1 and LPT2 in the table can be used. Also refer to the manual of the PC.

SMC_DRV2 and SMC_DV2A:

For the output of the signals to the stepper motor card only the address of the parallel port has to be assigned to the variable _LPT at the program start up. The other variables does not exist in this divers. Furthermore the function Init_Timer for initialising the interrupt timer should be executed.

b.) Function call **Referenz**
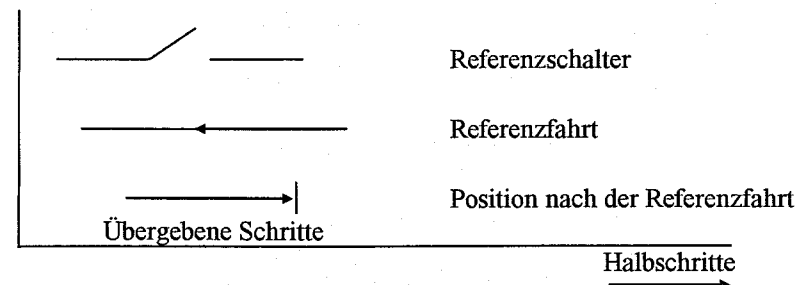
Referenz (xR_Step, yR_Step, zR_Step)

The variable types are int or integer.

The function does a reference positioning of the system. The function call includes as local parameters the number of half steps for each axis.

By using the function the stepper motor card does a slow movement in negative direction. After reaching the reference switch the movement changes the direction. Now the motors move the number of half steps from the parameters (xR_Step, yR_Step, zR_Step). So the system comes into a start up position. When the value for a parameter is 0 the reference positioning for the corresponding axis will not be done.

Example:

Referenz (200, 0, 0) does the reference positioning only in X-direction. The X- and the Y-axis will not move.

Referenzschalter

Referenzfahrt

Position nach der Referenzfahrt

Übergebene Schritte

Halbschritte

c.) Function call **Tastatur**

Tastatur (F_MODE)   F_MODE: variable type unsigned char or byte.

This function is for controlling the stepper motor card by keys.

 X-Axis: left/right
 Y-Axis: up/down
 Z-Axis: Page up/down
 Exit: ESC

F_MODE is for setting the mode for the keys described as follows:

Plotter-Mode:
F_MODE = 1 :   Page up = Z-axis current on
     Page down = Z-axis current off
3-Axis-Mode:
F_MODE = 0 :   Page up = Moving motor into positive direction
     Page down = Moving motor into negative direction

d.) Function call **Vecout**

Vecout (xStep, yStep, zStep)

xStep, yStep and zStep have the variable type int or integer.

This function call is for sending vectors to the stepper motor card.

The output speed depends on the speed of the PC. The right value must be found by trying.

For controlling of the stepper motor speed in the driver *SMC_DRV1* the variables Rampe_Start (start up frequency of the motors) and Rampe_Ende (working frequency of the motors) have to be set in the main program. For a safe ramping up of the motors the value for Rampe_Start must be higher than the value for Rampe_Ende.

The acceleration ramp for the motors can be adjusted by the variable *Steilheit*. This value also has to be found by trying.

For controlling of the stepper motor speed in the drivers *SMC_DRV2* and *SMC_DV2A* the variables V_Startfreq, V_Arbeitsfreq, Beschleunigung and Bremsen have to be set in the main program. The value is in steps per second and is independent of the speed of the PC. The value for the variable V_Startfreq must be smaller than the value for the variable V_Arbeitsfreq.

3.3 HPGL-Converter

With the HPGL-Converter it is possible to run plot files in the HPGL format (HP7475 - Plotter) with the stepper motor card (plotter mode). So it is possible to use standard CAD software for programming and running vectors.

In practice the following steps has to be done:

- Construct a drawing with a standard CAD - Program
- Save in the HPGL-format
- Start CAD-Output program and run the file with the stepper motor card.

The HPGL-Converter is not part of the stepper motor card software. It is obtainable in separate.

4. Data Interface

4.1 Connector Description of the Data Interface

a.) Parallel port (Centronics)

Pin-Number	Signal	Description
	SMC800/SMC1500 *without additional PCB*	SMC1500 with *additional PCB*
1	STROBE	STROBE
2	DATA 1	Clock signal X-Motor
3	DATA 2	Direction signal X-Motor
4	DATA 3	Clock signal Y-Motor
5	DATA 4	Direction signal Y-Motor
6	DATA 5	Clock signal Z-Motor
7	DATA 6	Direction signal Z-Motor
8	DATA 7	not connected
9	DATA 8	not connected
10	not connected	not connected
11	Input reference switch	Input reference switch
12,13	not connected	not connected
14	not connected	Relay - contact on/off
15	not connected	not connected
16	Ground	Ground
17,18	not connected	not connected
19-30	Ground	Ground
31,32	not connected	not connected
33	Ground	Ground
34,35,36	not connected	not connected

b.) Connector (only for SMC1500 with additional PCB)

Pin-Number	Description
1	Clock signal X-Motor
2	Direction signal X-Motor
3	Clock signal Y-Motor
4	Direction signal Y-Motor
5	Clock signal Z-Motor
6	Direction signal Z-Motor
7	Relay – front contact on/off (coil)
8	Relay – Ground (coil)
9	Relay – front contact connection 1
10	Relay – front contact connection 2

4.2 Signal Description of the Data Interface

a.) Parallel port (only for SMC800 and SMC1500 without additional PCB)

Signal	Description
STROBE	Receive Data (low active)
DATA1/DATA2	Current control for motor driver (Phase A)

DATA 1	DATA 2	
1	1	0% of max. current
0	1	20% of max. current
1	0	60% of max. current
1	0	100% of max. current

Signal	Description
DATA 3	Direction of current (Phase A)
DATA4/DATA5	Current control for motor driver (Phase B)
	Refer to DATA1/DATA2
DATA 6	Direction of current (Phase B)
DATA7/DATA8	Motor driver selection

DATA 7	DATA 8	
0	0	Motor driver X
1	0	Motor driver Y
0	1	Motor driver Z

5. Technical Data

	SMC800	SMC1500
Power supply:	14V ... 28V DC	24V ... 28V DC
Current:	max. 5A	max. 9,5A
Current per phase:	5mA ... 800 mA	5mA ... 1500mA
Switch output:		24V/8A AC (only with add. PCB)
Reference switch:	make contact	make contact
Input level:	TTL compatible	TTL compatible (Centronics) TTL and CMOS compatible (only with additional PCB)
Input frequency:	max. 5kHz	max. 5kHz

Schrittmotorsteuerkarte
SMC800 / SMC1500

Deutsch / Englisch

Technisches

Handbuch