

Manual



SMCI-46 Positioning Control (Including Output Module)

Contents

	Page
1	General -3-
2	Inputs/outputs and functions -4-
	2.1 Inputs -6-
	2.2 Outputs -7-
	2.3 Pushbuttons -7-
	2.4 Switches -8-
	2.5 Serial port -8-
	2.6 Displays -9-
3	Start-up -9-
4	SMCI 46.1 programming software -10-
	4.1 General -10-
	4.2 Installation -10-
	4.3 Starting the program -10-
	4.4 Setting the port address -11-
	4.5 Setting output module parameters -12-
	4.6 Programming the routes -13-
	4.6.1 Definitions -13-
	4.6.2 Programming a record -14-
	4.6.3 Entering a program -18-
	4.7 Manual run -20-
	4.8 Reference run -21-
	4.9 File menu -21-
5.	Manual reference run -22-
6.	Programming via the serial port -23-
	6.1 Parameter definitions -23-
	6.2 Data transfer -25-
7.	Error messages -33-
8.	Technical specifications (dimensions) -33-
9.	Radio interference -33-
10.	Dimensions -34-
Appendix	
A	Example programs for S7-215 -35-

1. General

Due to the integration of the components stepper-motor output module and positioning control, the SMCI-46 features a complete solution for the simple and quick start-up of 2-phase stepper motors in bipolar mode. The performance parameters of the output module and the program data of the positioning unit can be set and modified via the serial port. The "SMCI 46.1" programming software (for Windows 95/98 and NT) presents users, with and without programming skills, with a simple option for clearly and quickly designing up to 20 different routes and to transfer them to the control. Furthermore, the programmed "routes" can be started individually or automatically in succession from the program interface (such as for test purposes or for straightforward PC applications). Transferred data are stored in the SMCI-46 so that stand-alone operation is possible with the inputs and outputs. Experienced programmers are in position to design their own programs and thus customise the control to satisfy requirements.

2. Inputs/outputs and functions

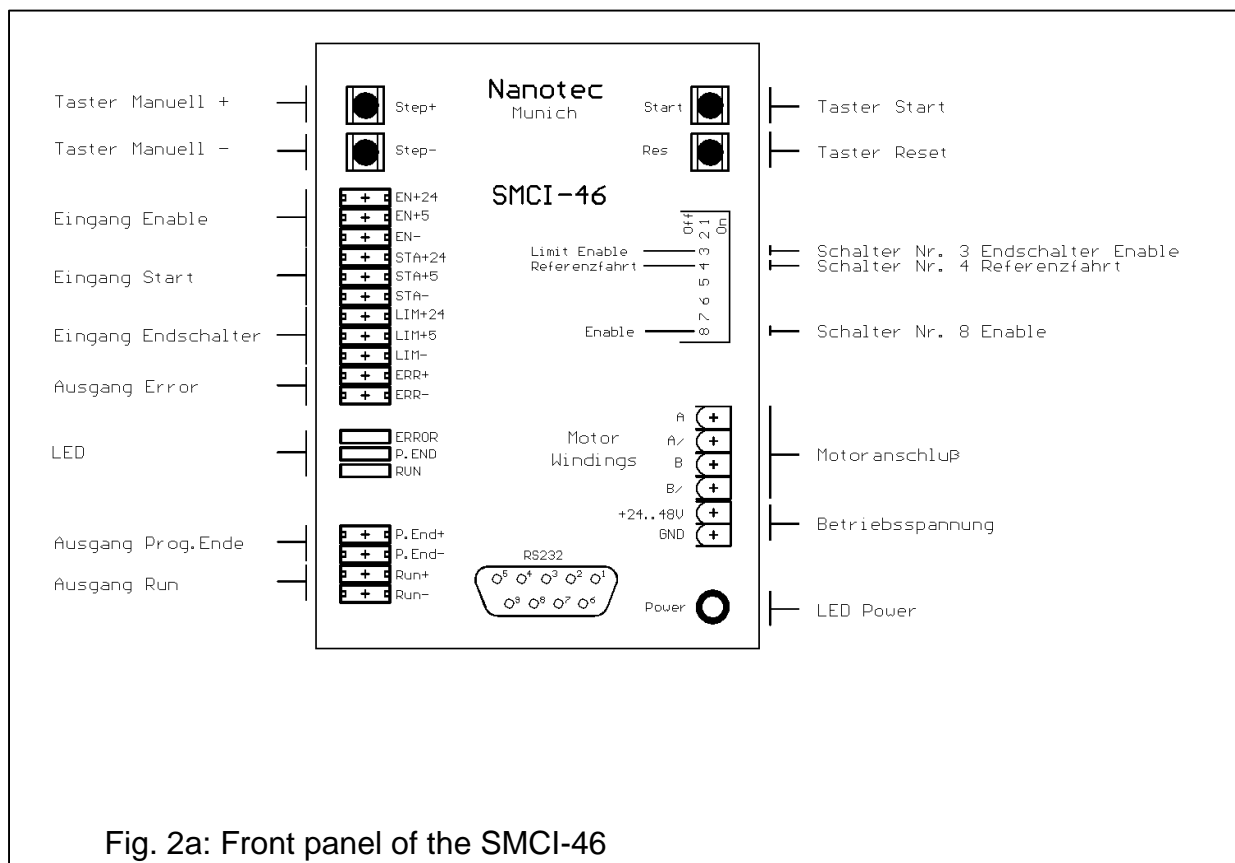
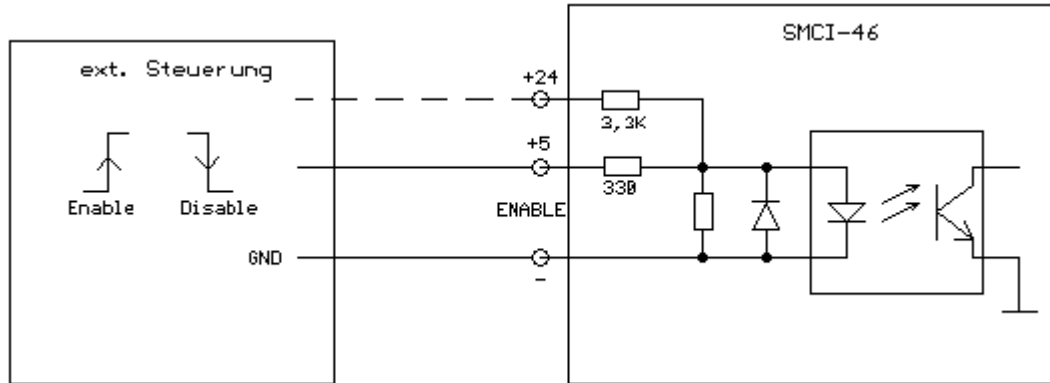


Fig. 2a: Front panel of the SMCI-46

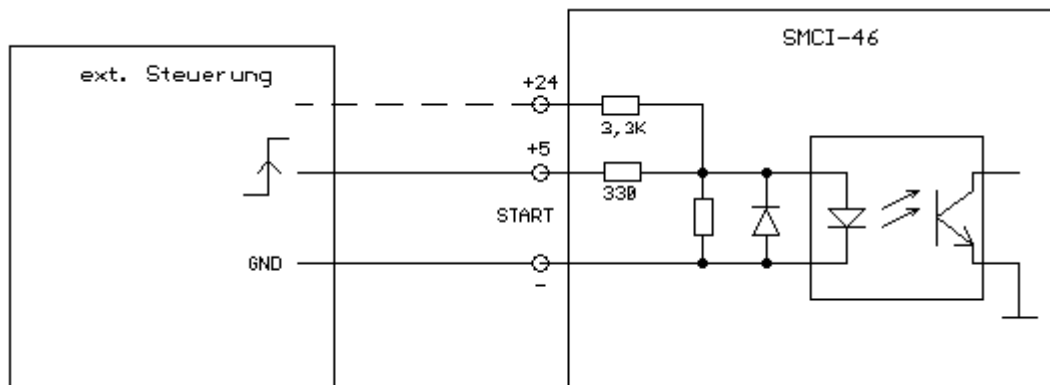
2.1 Inputs

a) Enable input



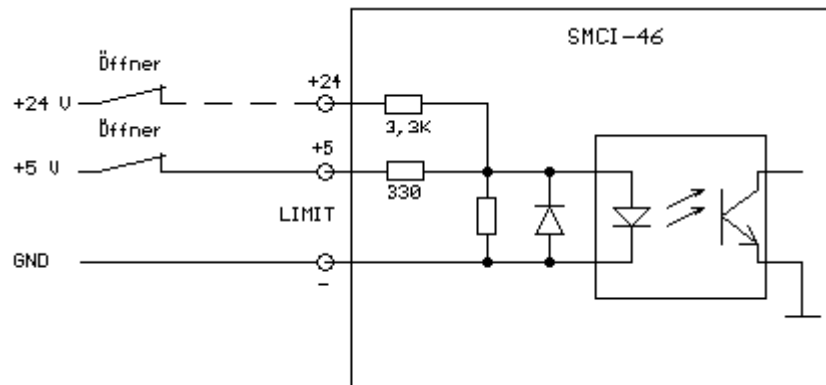
The Enable input can be used to switch the motor current on and off. After re-starting, the phases are subjected to the same current values as were applied immediately before the motor was switched off. If the Enable input is not connected, the motor current is switched on and off by means of the Enable switch (refer to 2.4c). The motor current can be switched off only in the end-of-program state (refer also to 2.2b).

b) Start input



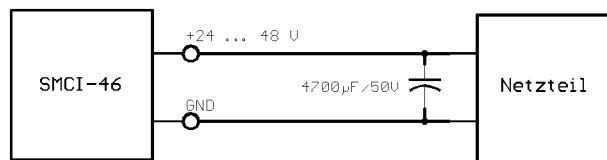
The Start input starts the program or the current record (refer to program start, external start). While a program or record is being executed (the LED Run lights), the input is ignored and therefore has no effect on the control.

c) Limit input (limit switch)



If the Limit input is disabled (switch open), the SMCI-46 immediately interrupts the current program and issues an error message (refer to 7.1). If no limit switch is connected to the control, switch 3 must be in the "On" position (refer to 2.4a), since it will not otherwise be possible to start a program.

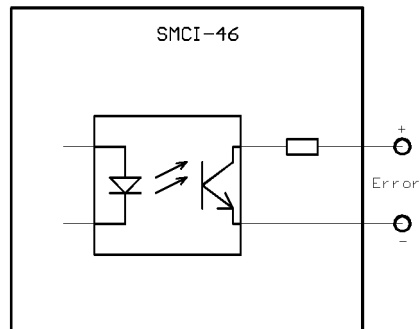
d) Operating voltage



The permissible operating voltage of the SMCI-46 is within the range from 24 to 48 V DC. A charging capacitor of at least 4700 µF/50 V (available as an accessory) **must** be connected to the supply in order to prevent the permissible voltage from being exceeded during a braking operation. **Reversal of the inputs/outputs can destroy the output module.**

2.2 Outputs

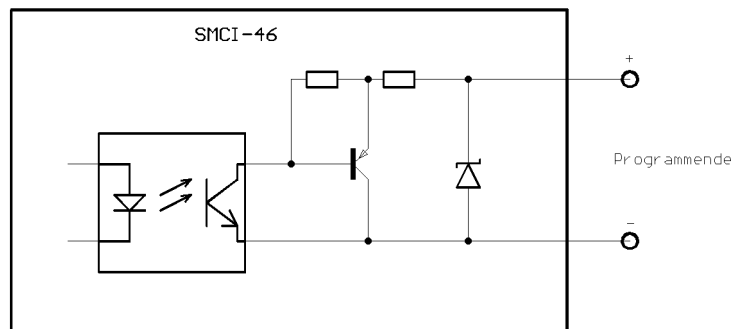
a) Error output



The Error output is enabled when the following events occur:

- Temperature rise:
The output module overheated and was switched off.
- Short-circuit:
There is a short-circuit between the motor phases or to GND. The output module was switched off automatically.

b) P.End (end-of-program) output

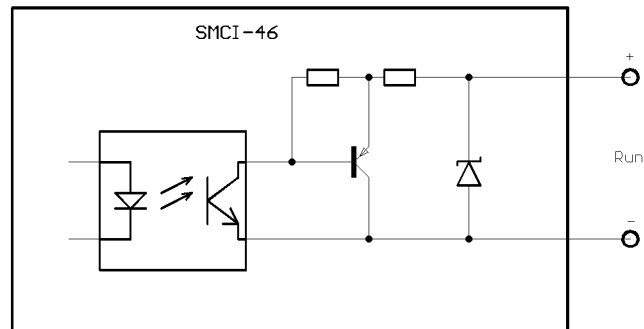


The output is enabled (the LED P.END lights) when the SMCI-46 is ready:

- for reference run
- for manual run
- for program start
- for programming

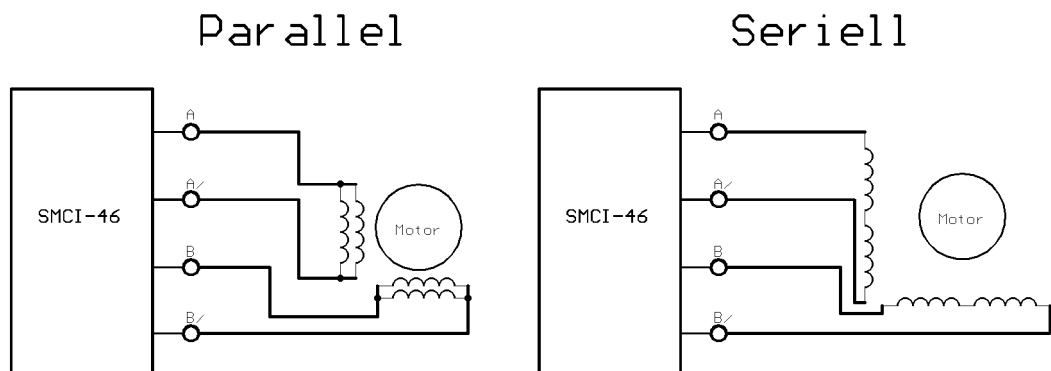
During a reference run, the output is active together with the Run output (refer to 2.2c). If the motor runs into a connected limit switch while a program is being executed, the output is pulsed together with the Run output at a frequency of approximately 3 Hz (refer to 7.1).

c) Run output



The Run output is active while a run command is being read out to the output module (without guarantee of motor rotation since an open control loop is involved). Furthermore, the output is enabled together with the P.End output, as described in 2.2b.

d) Motor connections



Refer to the data sheet of your stepper motor for the designations of the connecting leads. Check the connections carefully, **a wrong connection may result in destruction of the output module.**

2.3 Pushbuttons

a) Pushbuttons T+ and T-

The motor can be moved manually with the help of pushbuttons T+ and T- (the direction of rotation depends on how the motor phases are connected). The pushbuttons must additionally be operated for the manual reference run (refer to 5).

b) Start pushbutton

Operation of the Start pushbutton starts the currently loaded program (refer also to 2.1 c: Start input)

c) Reset pushbutton

Operation of the Reset pushbutton resets the control, and all inputs and outputs, as well as the displays and pushbuttons, have no function, and the motor current is switched off. When it is released, the SMCI-46 is reinitialised, and current is again applied to the motor, which assumes its initial position (home position).

2.4 Switches

a) Limit Enable (switch 3)

If no switch (NC contact) is connected to the input intended for it (refer to 2.1d), this switch must be in the "On" position. If the control has to poll the (connected) "NC contact", switch 3 must be set to "Off" (refer also to 2.1d: Limit input)

b) Reference run (switch 4)

If this switch is set to "On", the SMCI-46 goes to Reference Run mode (refer to 5.).

c) Enable (switch 8)

The Enable switch can be used to switch the motor current on and off. After re-starting, the motor windings are subjected to the same current values as were applied immediately before the motor was switched off. The motor current can be switched off (see also 2.1b): Enable input) only in the end-of-program state (refer also to 2.2b).

2.5 Serial port (RS232)



With the help of the serial port, the SMCI-46 can be connected to the PC or other devices capable of serial communication by means of a 9-pin D-Sub computer connecting cable (available as an accessory). The control can be programmed, started and set by means of the interface (refer to 6).

PIN assignment on the device:	PIN2	⇒	output Tx
	PIN3	⇒	input Rx
	PIN5	⇒	GND

2.6 Displays

a) Power LED (green)

The LED lights when the control is switched on.

b) P:End LED (yellow)

The LED lights when the End-of-Program output is active (refer to 2.2b).

c) Run LED (green)

The LED lights when the Run output is active (refer to 2.2c).

d) Error LED (red)

The LED lights when the Error output is active (refer to 2.2a).

3. Start-up

Perform the following steps to start the SMCI-46:

- 1.) Connect the motor to the designated outputs (refer to 2.2d).
- 2.) Connect the power supply (including the charging capacitor) to the terminals designated for it (refer to 2.1e).
- 3.) Set switch 3 (Disable limit switch) to "On", and set all other switches to "Off".
- 4.) Switch on the operating voltage, whereupon the LEDs Power and P.END will light. The red "Error" LED comes on briefly following power-up. Should the red LED not go out, turn off the control again immediately and check the connections.
- 5.) The SMCI-46 is then ready for operation, and you can use pushbuttons T+ and T- to move the motor manually (test run).
- 6.) After successful start-up, you can perform the other connections (such as the serial port, PLC, etc.) with the control switched off, and then switch on the SMCI-46 again.

4. Programming software

4.1 General

With the help of the *SMCI46.1* software (obtainable as an accessory) you can set and program the SMCI-46 on conventional PCs having Windows 95/98 or NT as the operating system via the serial port (RS232). The software allows access to the settings "phase current" and "Step mode" of the integrated output module and also the input of up to 20 different run commands, which can be started separately or interrelated so as to form a program. The stepper motor can also be operated in the "Manual Run" and "Reference Run" modes. All entries are performed on the PC keyboard.

4.2 Installing the software

1.) Software installation under Windows 95/98

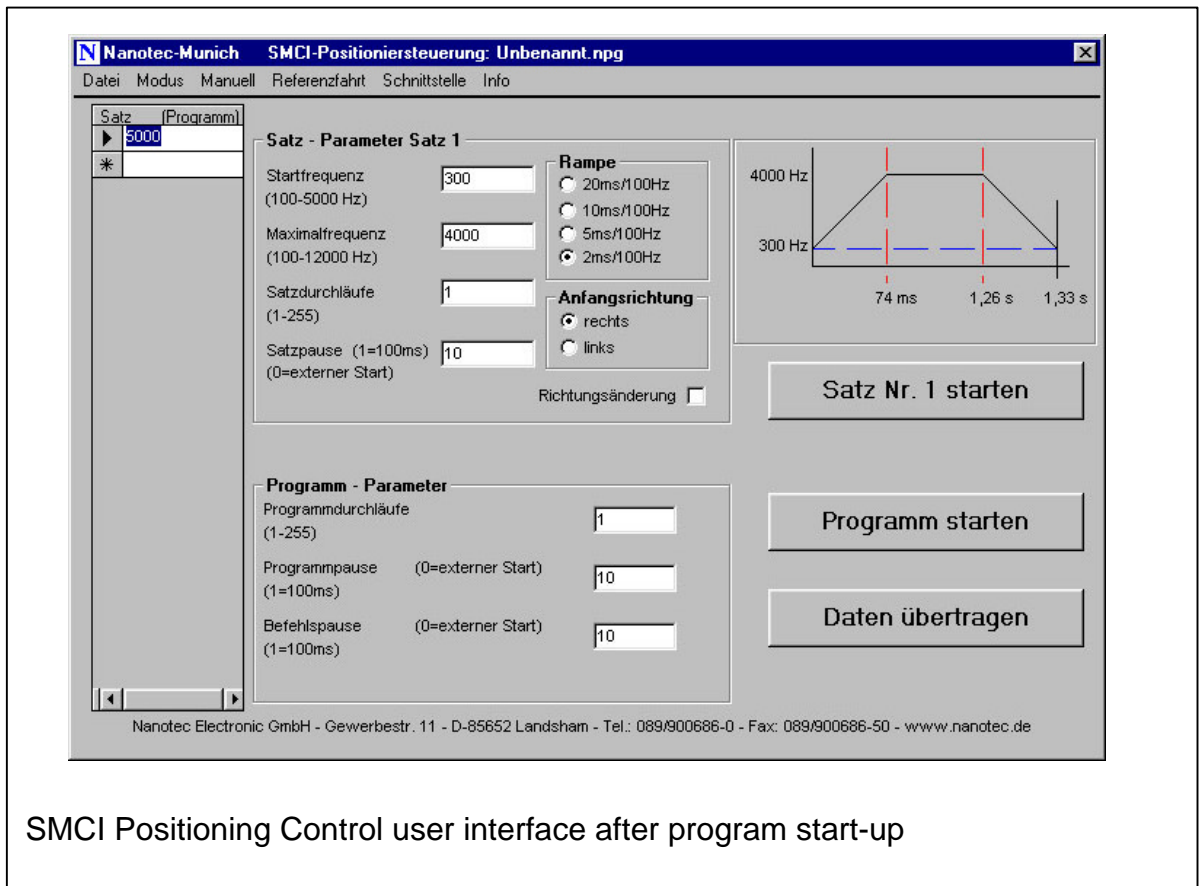
- a) Insert the *SMCI 46.1* CD-ROM in your CD-ROM drive.
- b) Should DCOM98 not be installed on your system already, open the directory called "Fixes" on the CD-ROM in Explorer and run the *DCOM98.exe* file. Then, follow the installation instructions.
- c) Open the directory called Setup on the CD-ROM.
- d) Run *Setup.exe* and follow the installation instructions.

2.) Software installation under Windows NT: (requirement: NT 4.0 plus Service Pack 5.0)

- a) Insert the *SMCI 46.1* CD-ROM in your CD-ROM drive.
- b) Open the directory called Setup on the CD-ROM.
- c) Then, run *Setup.exe* and follow the installation instructions.

4.3 Starting the program

After installing the program, you will find it on the Windows Start menu at *Programs P Nanotec Pro-Utility P SMCI 46.1*. Click the program with the mouse to load the *SMCI Positioning Control* user interface, the main user interface of the program. From this user interface you can access all the functions of the control. When the program starts up for the first time, it attempts to open serial port No. 1. Should it be assigned already, the program loads the Port Setting user interface. Then, perform the steps described in 4.4 "Setting the port address".



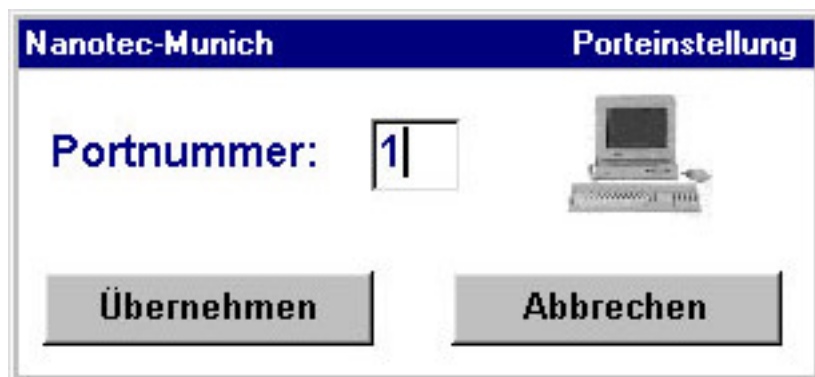
SMCI Positioning Control user interface after program start-up

4.4 Setting the port address

If you have more than one serial port and have not connected the SMCI-46 to the first port, you must change the port address. To do this, click with the mouse on "Port" on the menu of the program user interface.



The following dialogue box appears:



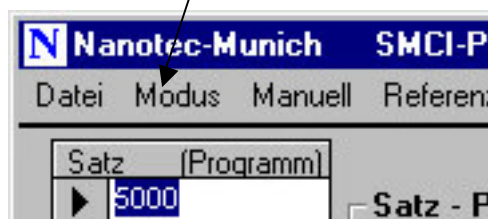
At "Port number", enter the number of the COM port to which you connected the control. Click on Apply with the mouse to save the new address and return to the main user interface.

If the program persists in Set Port mode after applying the port address, you have entered a port that is being used already, or the port address you entered was not correct. Enter a valid port number.

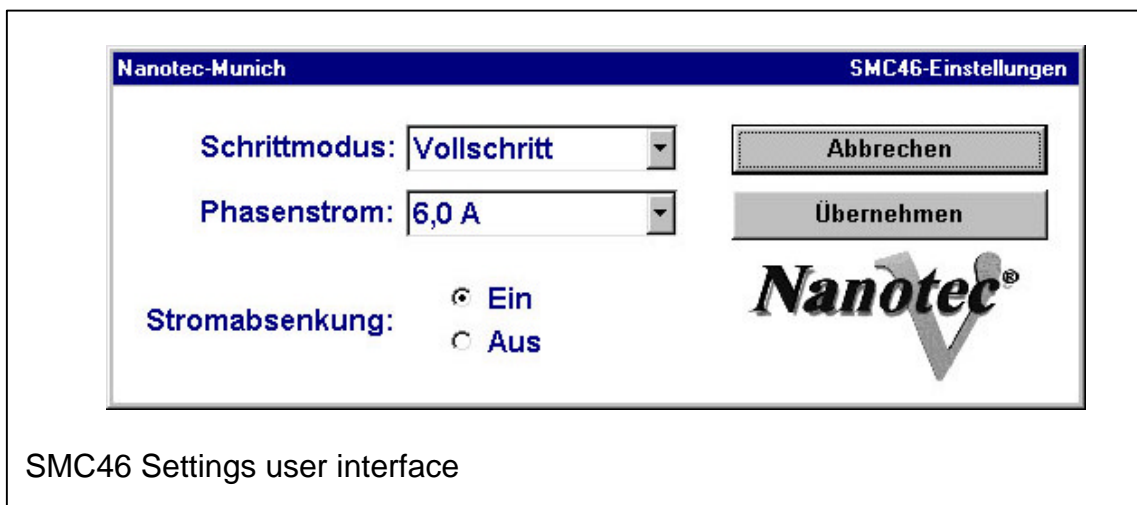
Click Cancel if you wish to return to the main user interface without changing the port number.

4.5 Setting output module parameters

Click with the mouse on "Mode" ⇒ "Parameters"



to set the parameters for the output module:



SMC46 Settings user interface

In the "Step Mode" drop-down list box you can select the following step modes:

- Full Step
- Half Step
- Quarter Step
- Fifth Step
- Eighth Step
- Tenth Step

In the "Phase Current" drop-down box you can select the following settings:

- 2.5 A
- 3.0 A
- 3.5 A
- 4.0 A
- 4.5 A
- 5.0 A
- 5.5 A
- 6.0 A

Furthermore, current depression is enabled or disabled at phase currents between 2.5 and 5 A, while it is always enabled at 5.5 and 6.0 A. This reduces the phase current by more than 30% when the motor comes to a rest for longer than 80 ms.

Click with the mouse on "Apply" to download the setting to the control and initialise it. Depending on the rotor setting, the motor can change its position in this case by up to 2 full steps.

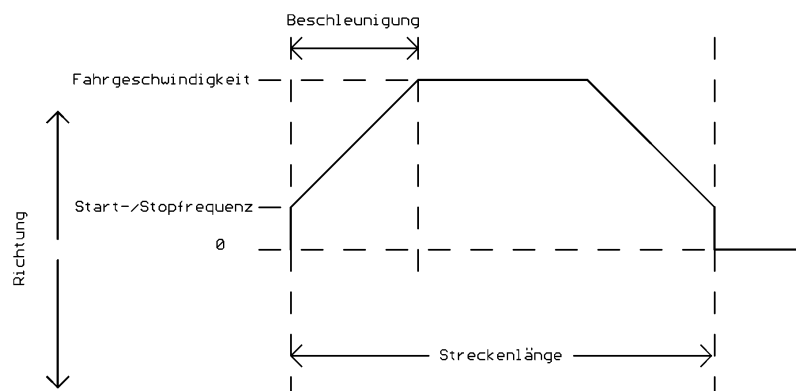
4.6 Programming the routes

Up to 20 different routes can be downloaded to the control with the help of the software. The programmed routes are saved and are still available when the SMCI-46 is switched on the next time. The old values are overwritten by reprogramming and can then not be called any more.

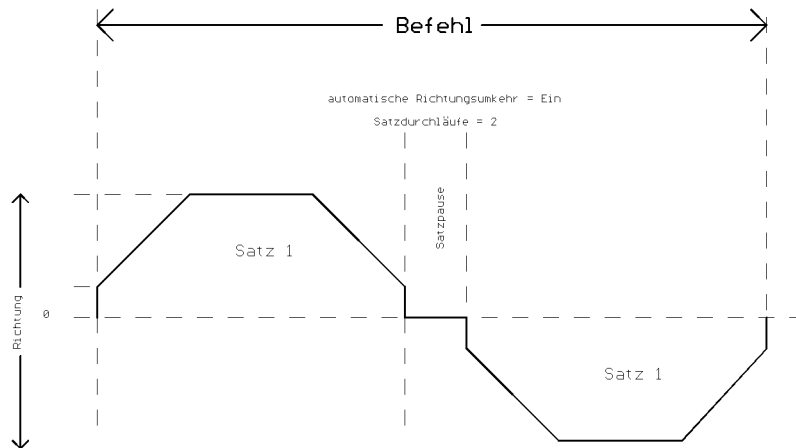
4.6.1 Definitions

Record: A record is a single route. A record consists of the following (record) parameters:

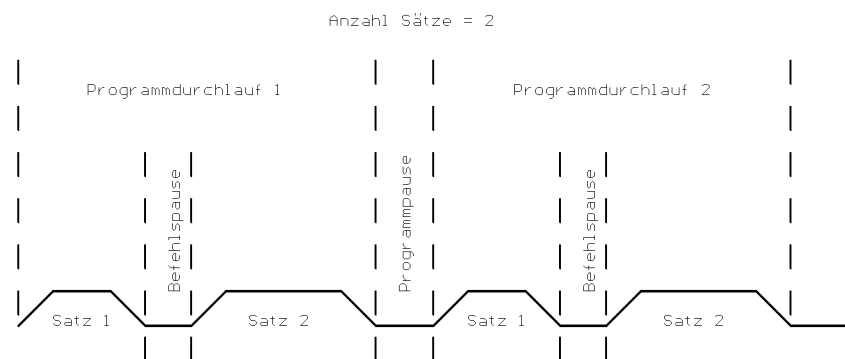
- of the section length [steps]
- of the start/stop frequency [Hz]
- of the running speed [Hz]
- of the direction [right, left]
- of the acceleration [ms/Hz]



Command: Apart from the record parameters, a command also consists of the details relating to repetitions of the record (record runs), the record idle time and the automatic change of direction.



Program: The program consists of up to 20 commands, and it contains the information about the idle time between the individual commands, the number of program runs and the program idle time (interval between program runs).

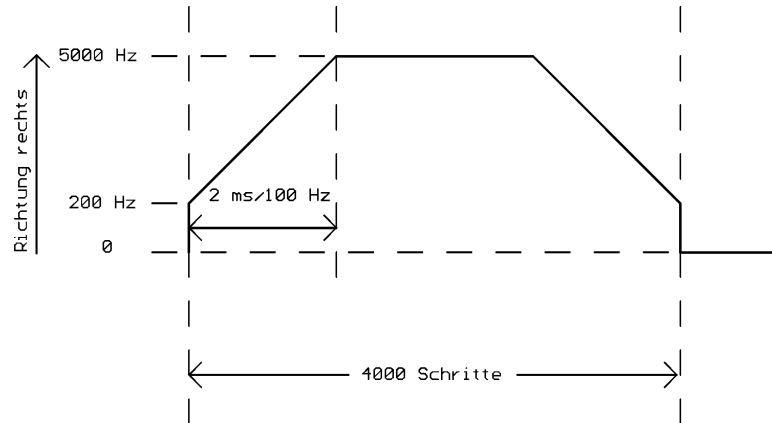


4.6.2 Programming a record

The input of a run command and all the details necessary for this will be illustrated by means of an example. At the end of the example, the SMCI-46 will be programmed and the record, and later the program, will be started. Go through this example and then modify the different parameters; it should become clear to you in a very short space of time how the program functions.

Example:

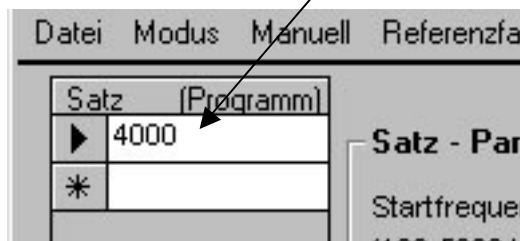
The stepper motor, after the program has started up, is required to run 4000 steps at a speed of 5 kHz to the right (the direction of rotation depends on how the motor is connected), while the start/stop frequency is 200 Hz. The quickest possible rate of acceleration is chosen.



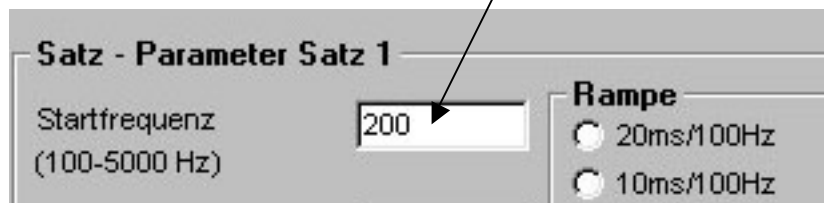
Example record

Input:

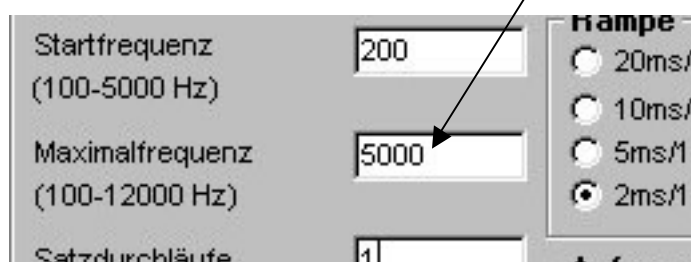
1. You want the motor to take 4000 steps
Using the mouse, enable the first field in the *Record (program) list* and enter 4000 as the value.



2. Start/stop frequency
In the *Start frequency* field, enter 200 for 200 Hz.



3. Travelling speed 5 kHz
In the *Maximum frequency* field, enter 5000 for 5000 Hz.



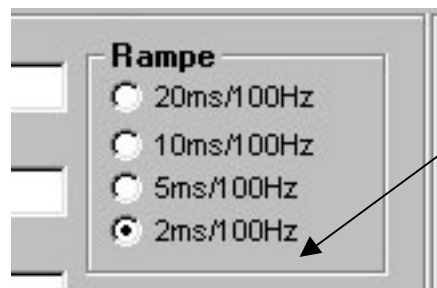
4. Direction

Using the mouse, select the *Right* check box at *Initial direction*.

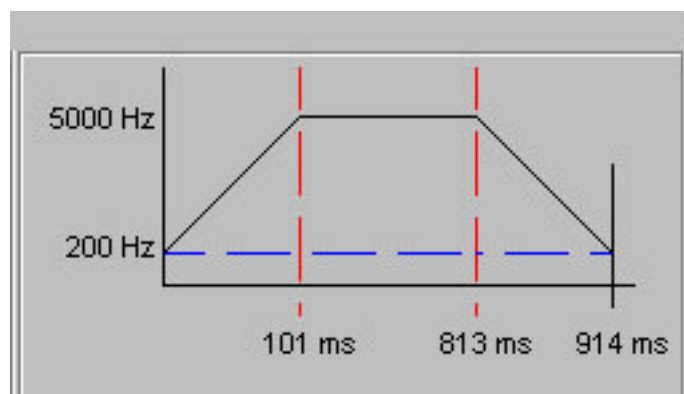


5. Acceleration

Using the mouse, select the *2 ms/100 Hz* check box at *Rampe*.

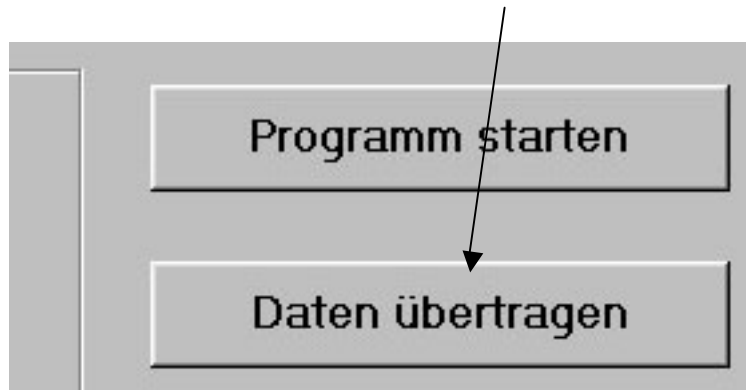


To the right of the parameters which you have just entered you will now see an image of the record you have just entered together with details of the duration of the acceleration phase, the starting time of the braking phase and the overall time of motion. For the time being, do not change the details referring to record runs, record pause, program runs, etc.



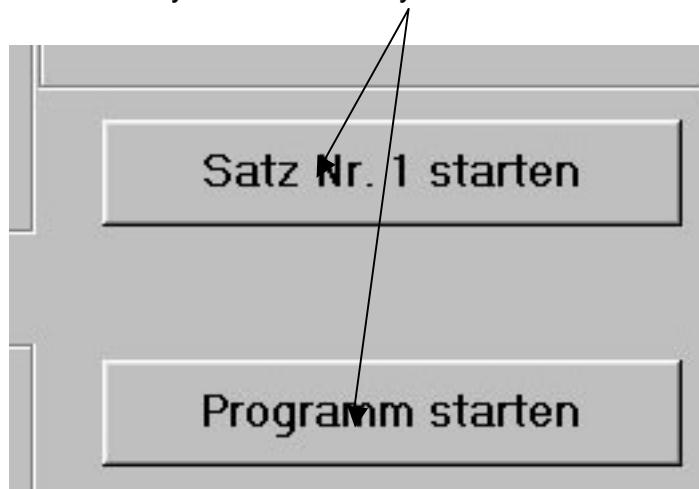
6. Data transfer

Click the *Download data* button. The values you set will now be downloaded to the control, the LED P.END goes out briefly during the download operation.



7. Record start

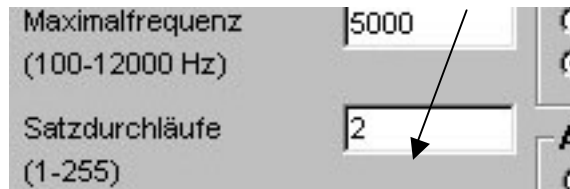
Once the SMCI-46 has been programmed, you can execute the record you have just entered by clicking the *Start record No. 1* button. The motor travels the distance you entered with the set output module parameters (Step mode and phase current). You can also start the record by clicking the *Start program* button, since the program currently consists of only one record.



4.6.3 Entering a program

You can now expand the record you entered in the above example to form a program:

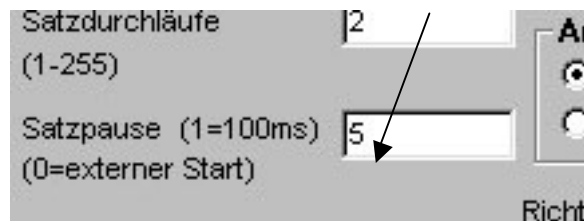
1. Change the value for the record runs from 1 to 2, then click the *Download data* button and restart the record (*Start record* or *Start program*).



⇒ The record will then be executed twice in succession. The interval between the record runs is 1 second.

2. The record interval is calculated as
input * 100 ms

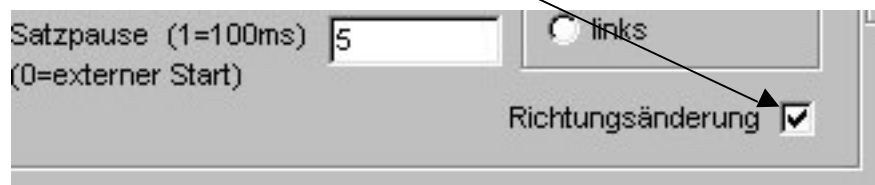
Change the value for the record interval to 5 (equivalent to 500 ms), download the data and execute the record again.



⇒ The record will be executed twice in succession. The interval between the record runs will then be 0.5 second.

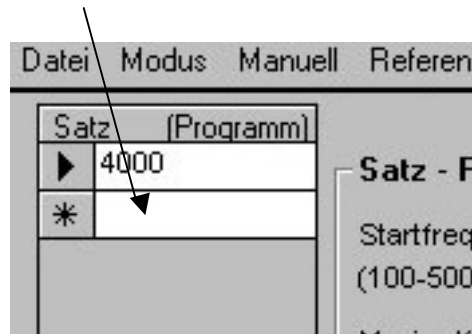
If you enter 0 instead of 5, each record run must be started externally (that is, by means of the Start input or the Start pushbutton).

3. Enable the "Change direction" field (then download data and start)



⇒ The direction of rotation is reversed at the start of the second record run.

4. Enter the second record
Enable the vacant field at *Record (program)* to enter the next record

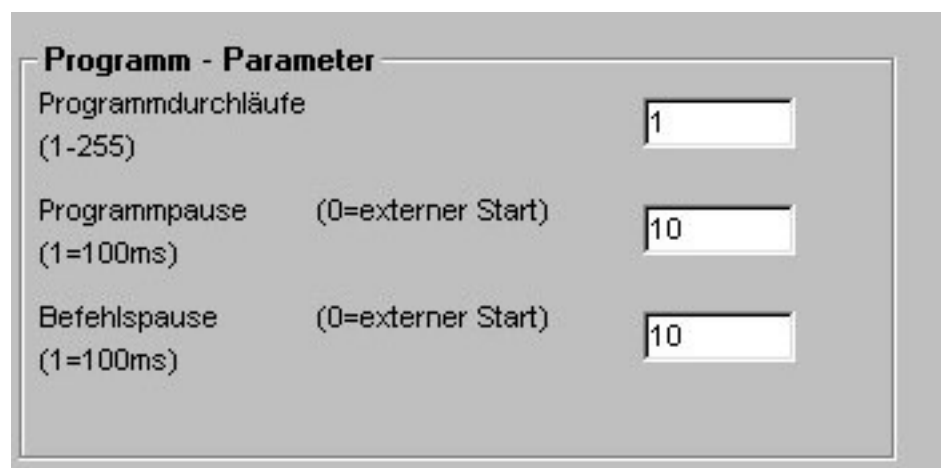


Then, enter any numbers of steps and modify the right-hand parameters of the record as you wish, as described in the example relating to the first record. Then, click the *Download data* button again.

The *Start program* and *Start record No.* buttons now have different functions.

Use *Start record No. 2* to start the record (record_2) which has just been enabled. If you move the mouse pointer into the first field at *Record (program)*, the button changes to *Start record No. 1* and you can only start the first record in this manner.

By using *Start program*, on the other hand, you can start both the records you entered in succession. At this point the program parameters are also important:



The command interval denotes the pause which is expected on the transition from record 1 to record 2. A "0" at this point signifies "external start" as with the record interval.

With the number of program runs, you specify how often the program (consisting of records 1 and 2) will be executed in succession. The program interval specified the duration of the

interval between program runs, where "0" again signifies external start.

4.7 Manual run

By choosing the *Manual* option from the menu you can move the motor manually into position with the mouse.



The "Manual run" user interface opens.



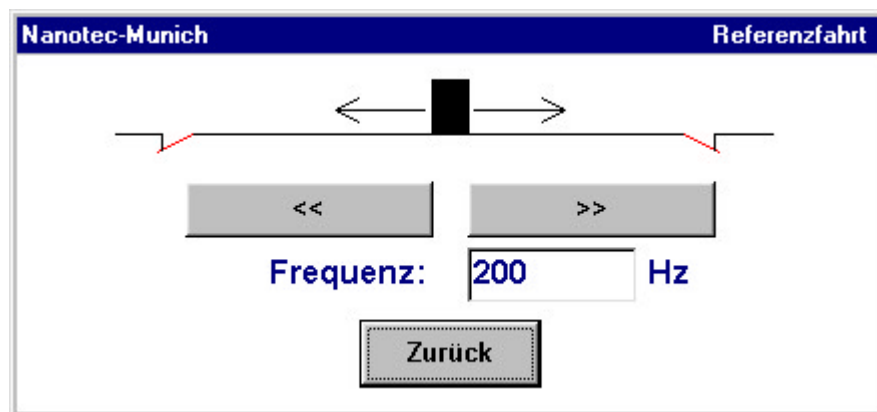
Click the < and > buttons (keep the left mouse button pressed down) to run the motor in Single Step mode. Five steps with a frequency of 2 Hz are output first. The motor then runs with a stepping frequency of approximately 30 Hz. Releasing the left mouse button stops the motor.

At 'Quick motion?', enter the travelling speed and the length of the section (*frequency and steps*). The motor starts with a fixed ramp, it runs and stops with the specified frequency, i.e. it does not run a braking ramp. For this reason the frequency should not be too high in this particular instance. The run stops after the specified number of steps.

Click *Back* to return to the main user interface.

4.8 Reference run

Choose the *Reference run* option from the menu to go to the "Reference run" user interface.



In the Frequency field, enter the stepping speed with which the motor should be driven in the direction of the reference point switch (Limit input connection, refer to 2.1c). Then, click the << or >> buttons (depending on the position of the limit switch) and keep the left mouse button pressed down. The motor now runs in the specified direction at the specified frequency. If the motor runs into the limit switch, the control automatically changes the direction of rotation and runs down from the limit switch at a slow frequency. When the motor stops, the reference run is completed, the motor now stands exactly step in front in front of the limit switch. The maximum frequency for the reference run is 600 Hz.

The reference run can be performed at any time. Click *Back* to return to the main user interface.

Caution: When a reference run is in progress, a limit switch must be connected to the Limit input and the Limit-En switch must be at "Off".

4.9 File menu

With the File menu you can save your program, open saved programs and exit the program, in the manner to which you are accustomed in other applications.

5. Manual reference run

Resetting switch No. 4 (refer to 2.4b) switches the SMCI-46 to Reference Run mode. When a limit switch is connected, switch 3 (Limit Enable, refer to 2.4a) must be in the Off position. If it not, the limit switch cannot be detected while the program is being executed.

Using pushbuttons T+ and T- you can now run the motor in the direction of the limit switch (refer to 2.1d for connecting it). If the motor runs into the limit switch, the control automatically changes the direction of rotation and runs down from the limit switch at a slow frequency. When the motor stops, the reference run is completed, the motor now stands exactly one step in front in front of the limit switch.

The reference run can be performed at any time. Upon completion of the reference run, return switch No. 4 to the Off position. The control changes to P.End mode.

6. Programming via the serial port

6.1 Parameter definitions

The routes of the motor consist of program and record parameters. Program parameters within this context are the details valid for the overall program, whereas the record parameters define the different routes which, as a whole, constitute the program.

Program parameters:

a) *Program runs* (value 1 to 255)

An integer which specifies how often the program should be executed in succession.

b) *Program interval* (value 0 to 255)

An integer which specifies how long the interval between the different program runs is (interval = integer * 100 ms). A "0" disables automatic program start and enables the external start.

c) *Number of records* (value 1 to 25)

An integer which specifies how many records are contained in the program.

d) *Command interval* (interval between the different records, value 0 to 255)

An integer which specifies how long the interval is for the transition (change of command) from one record to the next (interval = integer * 100 ms). A "0" disables automatic start and enables the external start.

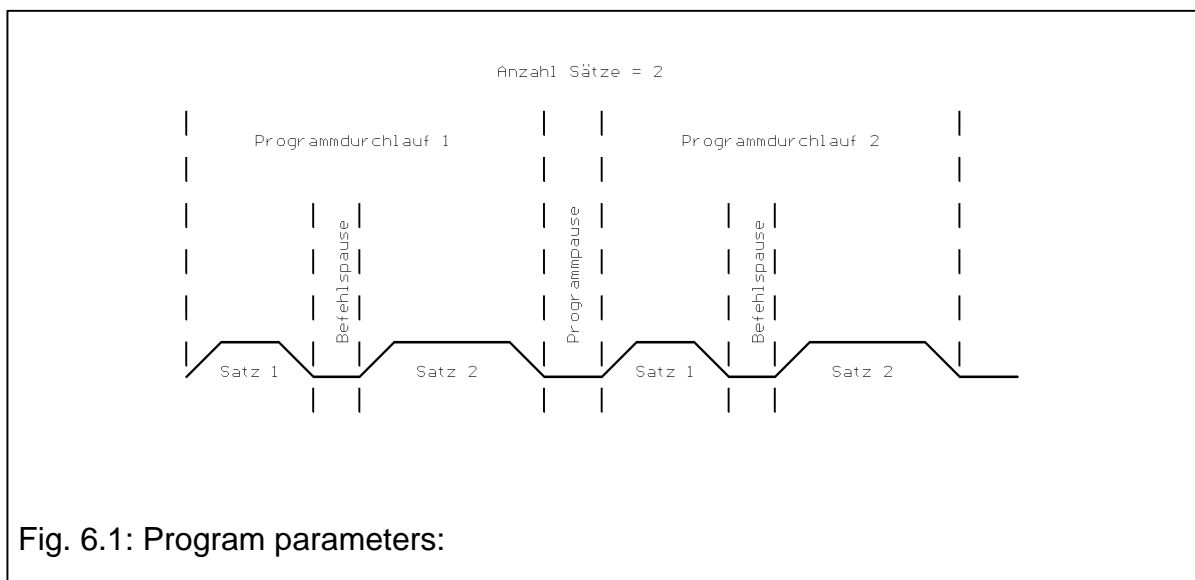


Fig. 6.1: Program parameters:

Record parameters:

a) *Steps* (value 1 to 16 million)

Number of output pulses (length of the section travelled) in 24 bit binary code.

b) *Start frequency* (value 1 to 239)

Start/stop frequency of pulse output (lowest point of the ramp)

Calculation: start frequency = frequency [Hz] / 50 - 1

- c) *Maximum frequency* (value 1 to 239, 1 = 100 Hz, 239 = 12 kHz)
End frequency of pulse output (top point of the ramp and travelling speed)
Calculation: maximum frequency = frequency [Hz] / 50 – 1
- d) *Record runs* (value 1 to 255)
An integer which specifies how often the record will be executed directly in succession.
- e) *Record interval* (interval between the different record runs, value 0 to 255)
An integer which specifies how long the interval is between record repetitions (interval = integer * 100 ms). A "0" disables automatic start and enables the external start.
- e) *Direction* (value 1 or 0)
Direction of rotation of the motor during the first record run.
- f) *Automatic reversal of direction* (value 1 or 0)
With several record runs, the direction of rotation of the motor is changed by enabling automatic reversal of direction (1 = enabled) upon each start of the record (automatic or external).
- g) *Ramp* (value 1 to 4)
The value specifies the slope of the ramp (motor acceleration).
1 = 2 ms/100 Hz
2 = 5 ms/100 Hz
3 = 10 ms/100 Hz
4 = 20 ms/100 Hz

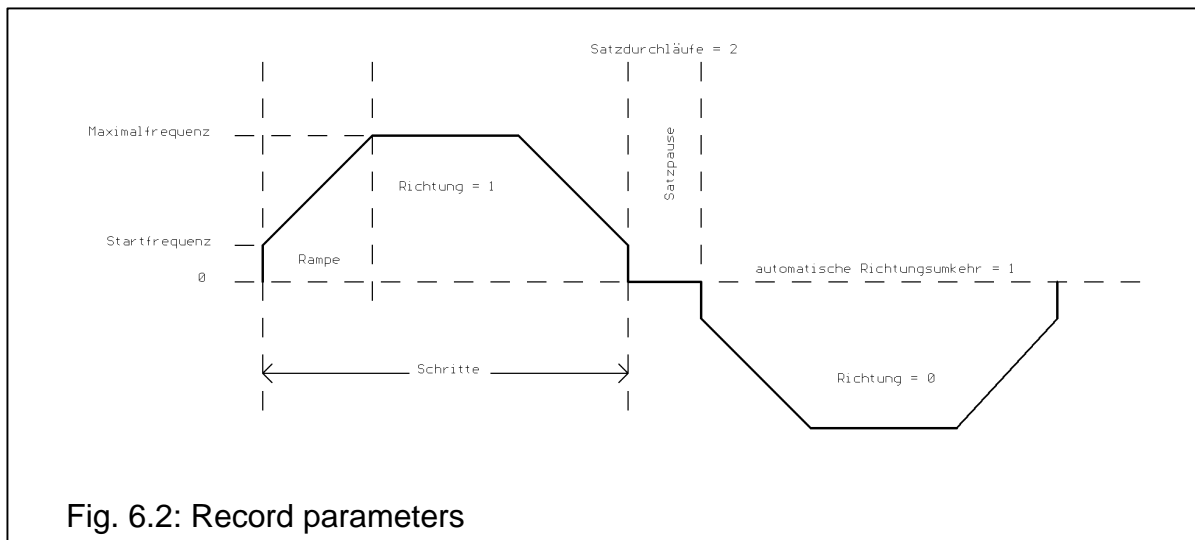


Fig. 6.2: Record parameters

6.2 Data transfer

Data are transferred with the following serial interface settings:

- Baud rate 19200
- No parity
- 8 data bits
- 1 stop bit

The different bytes are not transferred directly one after the other to the SMCI-46; an interval of at least 100 μ s has to be maintained between data transfer.

Caution: Programming is possible only when the control is in the "end of program" state.

Transferring program and record parameters

To switch the SMC46-X to Programming mode, a chr(03) is sent first. The program and record parameters are then transferred in the following order:

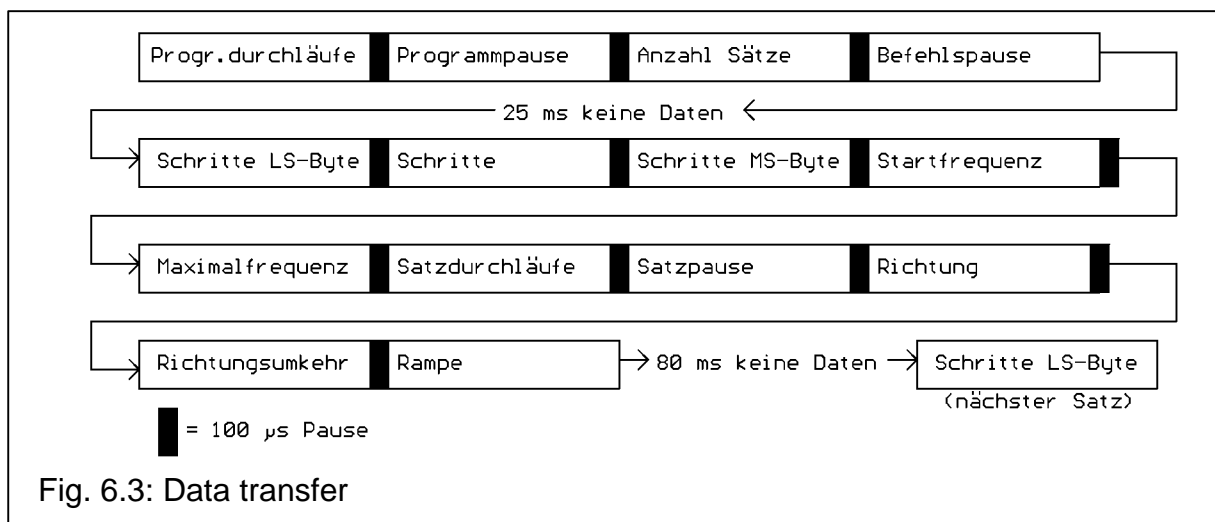


Fig. 6.3: Data transfer

If the program contains several records, an interval of 80 ms must be maintained between programming of the ramp of the preceding record and the step LS Byte of the next record (the data are saved). The program parameters, however, are transferred only once at the start of programming, irrespective of the number of records. The LED "P.END" goes out while programming is in progress.

Example (program with 3 records):

After the program has started, the stepper motor is required to take 4000 steps to the right at a speed of 5 kHz, then automatically perform 8 sets of 1000 steps alternately to the left and right, with an interval of 2 sec. between record runs and a speed of 200 Hz, and is then required to go back again to the start position with a speed of 3 kHz. The start/stop frequency is the same for all records at 200 Hz, and the motor travels with maximum acceleration. The operation has to be repeated 20 times. The different records and program repetitions are started externally.

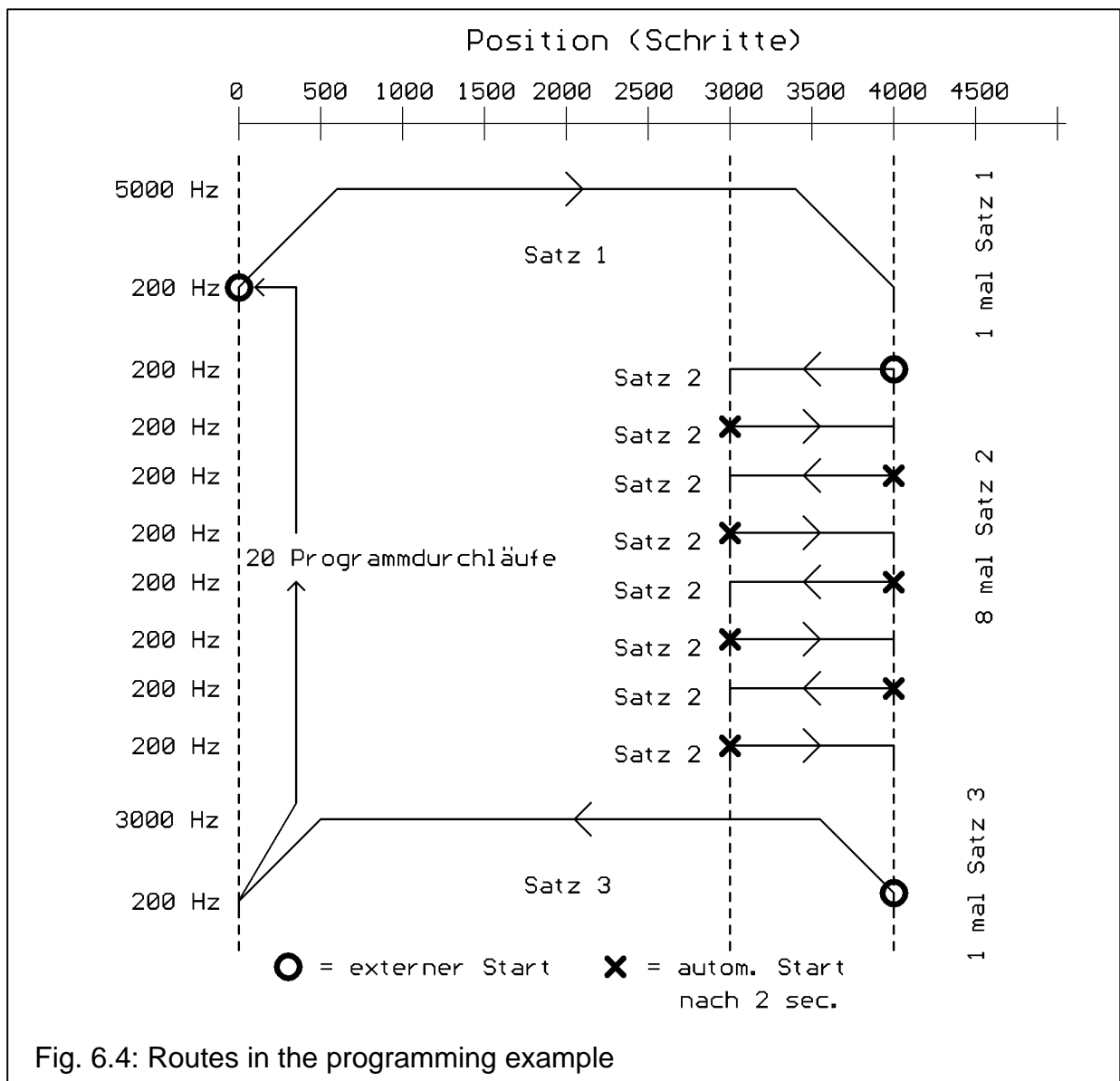


Fig. 6.4: Routes in the programming example

Defining program and record parameters

a) Program parameters:

- Program runs
The operation has to be repeated 20 times. \Rightarrow
Program runs = 20
- Program interval
The program runs are started externally \Rightarrow
Program interval = 0
- Number of records
The program consists of three records \Rightarrow number of records = 3
- Command interval
The records are started externally \Rightarrow command interval = 0

b) Record parameter, record 1

- Steps:
Record 1 contains 4000 steps \Rightarrow 4000 = 00000000 00001111
10100000 (binary)
 \Rightarrow LS byte = 160 (decimal)
Steps = 15 (decimal)
MS byte = 0 (decimal)
- Start frequency
The start frequency is 200 Hz \Rightarrow start frequency = $200/50-1 = 3$
- Maximum frequency
The maximum frequency is 5000 Hz \Rightarrow maximum frequency = $5000/50-1 = 99$
- Record runs
The record is run once only \Rightarrow record runs = 1
- Record interval
The record interval is not effective in record 1, since the record is only executed once \Rightarrow record interval = 1 (virtual value)
- Direction
Assumption: direction = 1 means Right (depends on how the motor is connected) \Rightarrow direction = 1
- Reversal of direction
Automatic reversal of direction is disabled \Rightarrow reversal of direction = 0
- Ramp
Maximum acceleration is set \Rightarrow ramp = 1

c) Record parameter, record 2

- Steps:
Record 2 contains 1000 steps $\Rightarrow 1000 = 00000000 \quad 00000011$
11101000 (binary)
 \Rightarrow LS byte = 232 (decimal)
Steps = 3 (decimal)
MS byte = 0 (decimal)
- Start frequency
The start frequency is 200 Hz \Rightarrow start frequency = $200/50-1 = 3$
- Maximum frequency
The maximum frequency is 200 Hz \Rightarrow maximum frequency = $200/50-1 = 3$
- Record runs
The record is run eight times \Rightarrow record runs = 8
- Record interval
The record interval is 2 seconds (automatic start) \Rightarrow record interval = $2 \text{ sec.}/100 \text{ ms} = 20$
- Direction
First start of the record, to left \Rightarrow direction = 0
- Reversal of direction
Automatic reversal of direction is enabled \Rightarrow reversal of direction = 1
- Ramp
Maximum acceleration is set \Rightarrow ramp = 1

d) Record parameter, record 3

- Steps:
Record 3 contains 4000 steps $\Rightarrow 4000 = 00000000 \quad 00001111$
10100000 (binary)
 \Rightarrow LS byte = 160 (decimal)
Steps = 15 (decimal)
MS byte = 0 (decimal)
- Start frequency
The start frequency is 200 Hz \Rightarrow start frequency = $200/50-1 = 3$
- Maximum frequency
The maximum frequency is 3000 Hz \Rightarrow maximum frequency = $3000/50-1 = 59$
- Record runs
The record is run once only \Rightarrow record runs = 1
- Record interval
The record interval is not effective in record 3, since the record is only executed once \Rightarrow record interval = 1 (virtual value)
- Direction
The record is run to the left \Rightarrow direction = 0
- Reversal of direction
Automatic reversal of direction is disabled \Rightarrow reversal of direction = 0
- Ramp
Maximum acceleration is set \Rightarrow ramp = 1

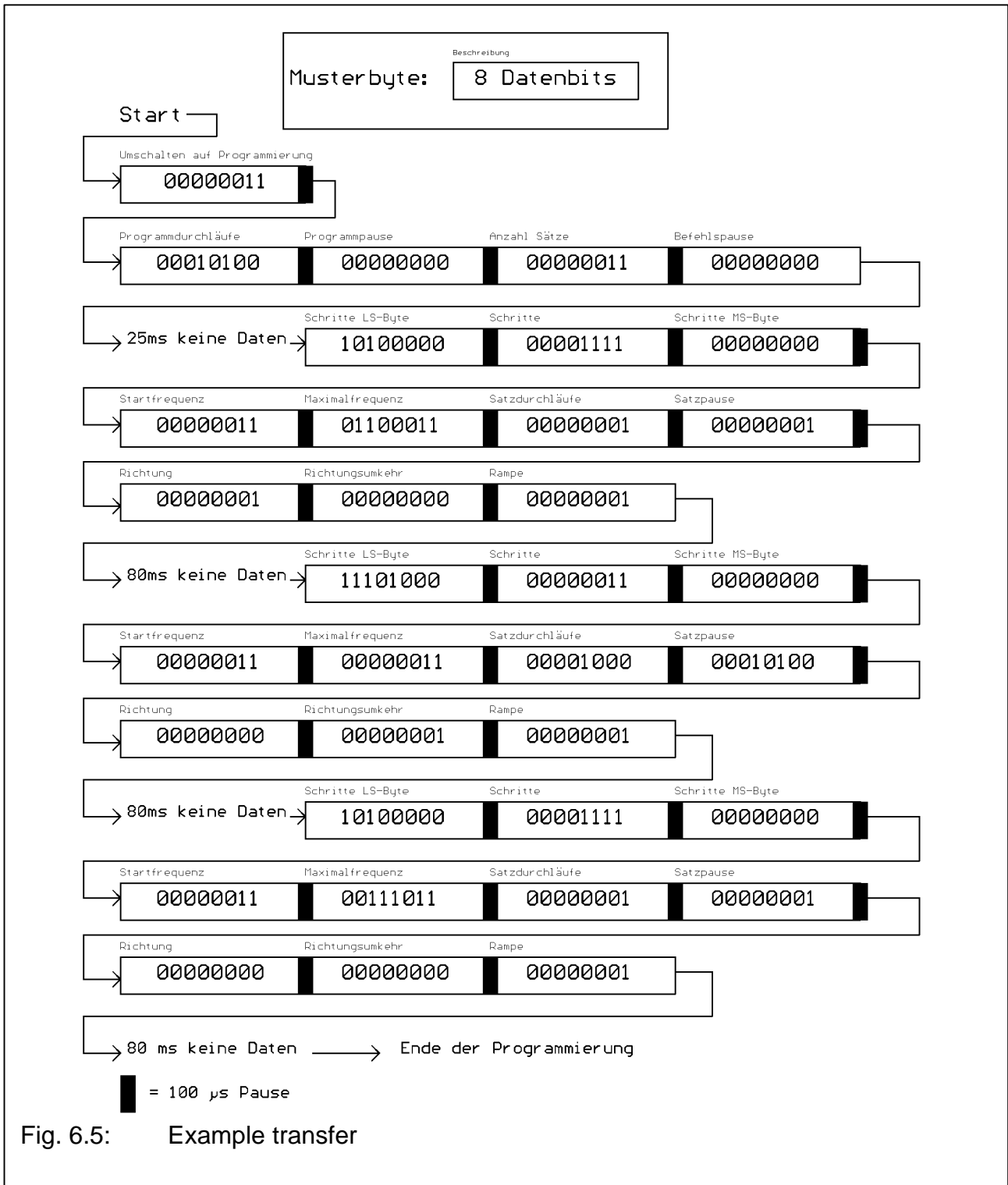


Fig. 6.5: Example transfer

Following successful programming, the LED "P.END" lights again, and the data are saved. The SMCI-46 is now ready for operation and the program can be run. The program data are still available after the control has been switched back on; they are only discarded by reprogramming the control.

Programming without saving the data in the EEPROM

When programming the SMCI-46, you get around saving the data in the EEPROM (used with continuously changing record parameters). With this type of programming, there are no program parameters; only the data for the record which has to be executed are transferred. The record parameters contain:

- Steps (consisting of 3 bytes)
- Start frequency
- Maximum frequency
- Record runs
- Record interval
- Direction
- Reversal of direction
- Ramp

The definition, type and calculation of the different bytes correspond to normal programming (with data being saved). The programmed record can be started in a conventional manner after programming has been completed; it is only executed once, however, and after that reprogramming becomes necessary. If the control is started again without prior programming, the data contained in the EEPROM are executed.

Programming without saving is initiated by means of a chr(7).

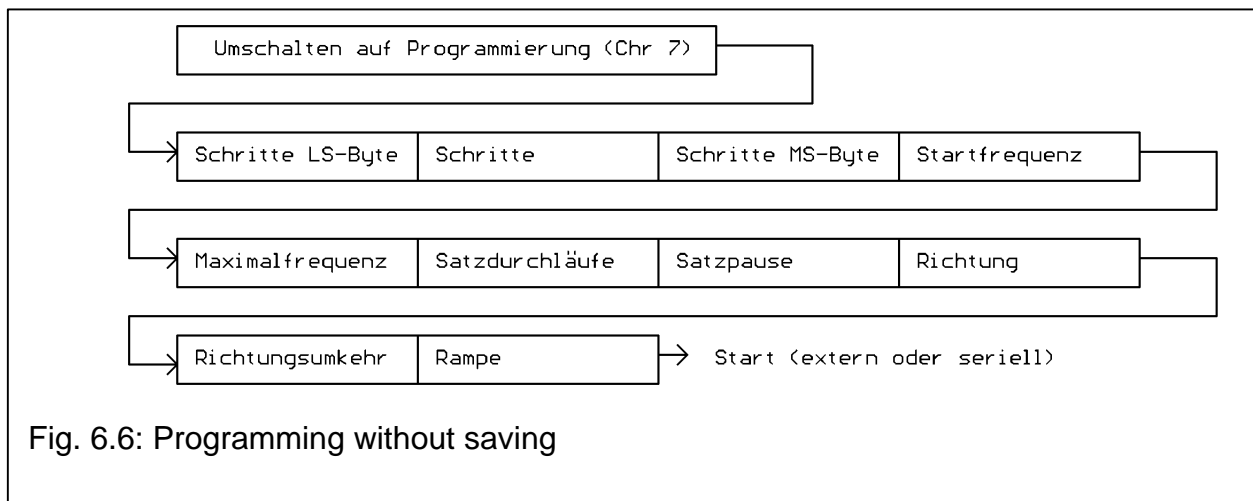


Fig. 6.6: Programming without saving

Starting the program

In addition to starting the program externally, the SMCI-46 can also be started with the help of the serial port. To accomplish this, a "04" is transferred to the control, which then triggers an immediate program start. Be sure that the SMCI-46 has been programmed with the necessary parameters beforehand.

Start → 00000100 → Programmausführung

Fig. 6.6: Program start via RS232

Starting the record

The different records can be called and run individually with the help of the serial port. Using this method of running, up to 25 records can be programmed (as described above or with SMCI-46 software (up to 20 records)) and called as and when required. To do this, a "05" is first sent to the control, followed by the required record number. The record is started, including record repetitions and intervals, after it has been transferred.

Start → 00000101 → Satznummer → Satzstart

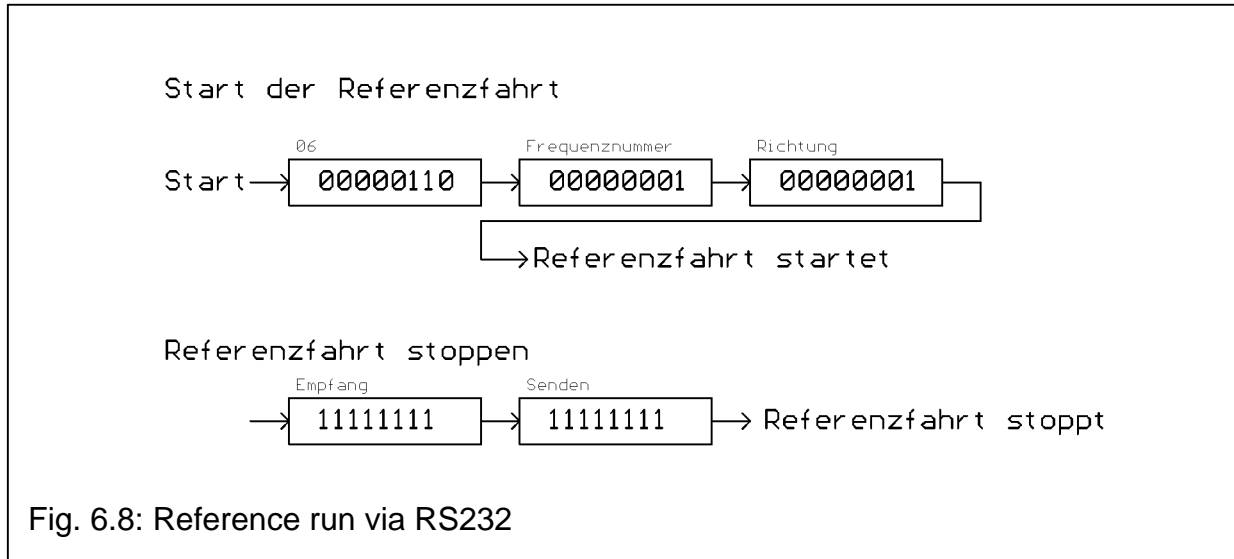
Beispiel: Start des 3. Satzes

Start → 00000101 → 00000011 → Start Satz 3

Fig. 6.7: Record start via RS232

Reference run

The reference run can also be called at any time in the *end-of-program* state (the LED P.END lights) via the serial port. To achieve this, first a "06" (go to Reference Run mode) and then the frequency number (frequency for the reference run, calculation: $(\text{frequency}/50)-1$) and the direction (00 or 01) are transferred. Output is started after the direction has been transferred. The SMCI-46 sends a 255 after every step as an identification for it being possible to stop output within the next interval (duration: $t \sim 1 / \text{frequency}$) by sending a "255".



Setting the step mode and phase current

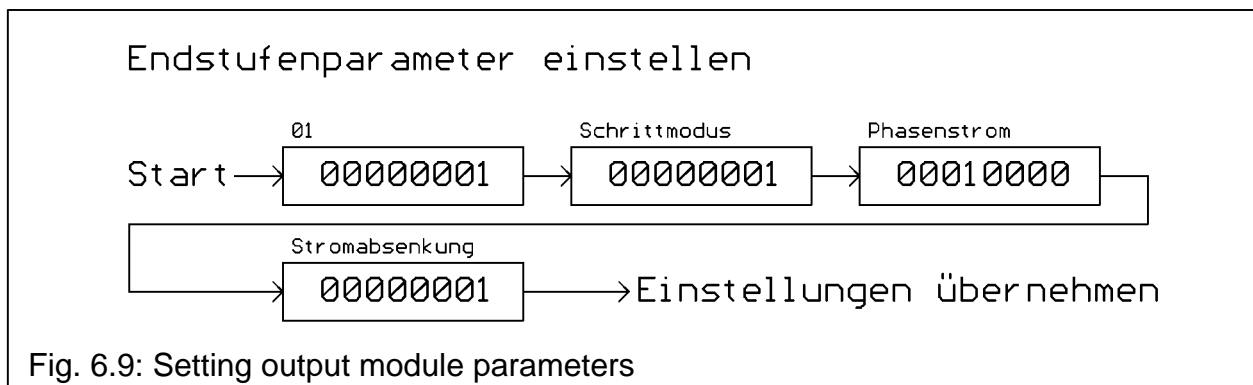
In all, four bytes are transferred to modify the step mode or phase current. The first byte ("01") sets the output module to *Set Output Module Parameters* mode. The desired step mode, phase current and current setting are then transferred to the SMCI-46 as shown in the table below.

Step mode	Byte
Full Step	00
Half Step	01
Quarter Step	02
Fifth Step	03
Eighth Step	04
Tenth Step	05

Phase current	Byte
2.5 A	1
3.0 A	2
3.5 A	4
4.0 A	8
4.5 A	16
5.0 A	32
5.5 A	64
6.0 A	128

Current depression	Byte
On	00
Off	01

Current depression is automatically enabled with a phase current of 5.5 and 6.0 A. In the example, the output module is configured with the settings *Half step*, *4.5 A* and *Current depression off*.



7. Error messages

7.1 P.End and Run flash uniformly

The control has run into a limit switch during program execution. Perform a reset and run the control manually from the limit switch with T+ or T-. Check for the possible cause of the error.

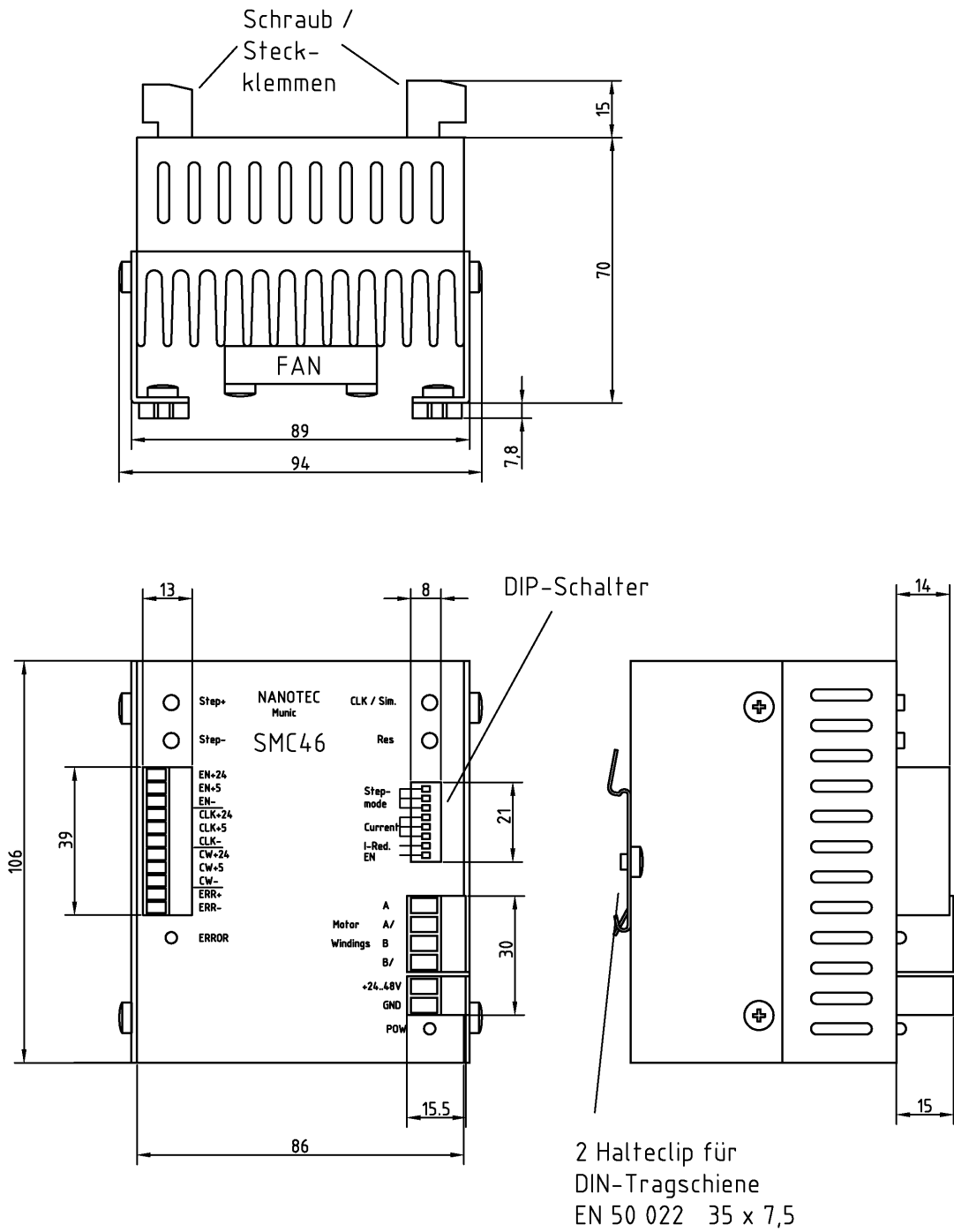
8. Technical specifications

Operating voltage	24 to 48 V DC
Phase current	2.5 to 6 A
Current setting	Via RS232
Duty type	Bipolar chopper driver
Operating modes	Full Step Half Step Quarter Step Fifth Step Eighth Step Tenth Step
Step setting	Via RS232
Stepping frequency	100 to 12000 Hz
Test mode	Using pushbuttons T+ and T-
Outputs	Optocoupler Output current not more than 50 mA Output voltage not more than 35 V
Inputs	Standard 5 V, 24 V optional Optocoupler 10 mA input current
Current depression	0 and 30% via RS232
LEDs	Power (operating voltage) Run (output in progress) P.End (ready for operation) Error (temperature rise, short-circuit)
Temperature range	0 to 40 °C
Type of connection	Pluggable screw terminals
Type of mounting	On DIN mounting rail Directly screwed

9. Radio interference

Electric and magnetic alternating fields are generated around conducting leads of the SMCI-46 on account of its method of operation; this is particularly true of the motor and supply lines. These fields may also interfere with other devices. At the same time, the SMCI-46 may be destroyed by similar fields generated by other devices.

10. Dimensions



Appendix A: Example programs for S7-215

```
//  
//Program start of the SMCI-46 via the serial port  
//  
  
NETWORK 1  
LD      SM0.1          //First scan cycle  
MOVB    5, SMB30       //19,200 bauds, 8 data bits, no parity  
MOVB    1, VB100       //1 character will be transferred  
MOVB    16#4, VB101    //Transfer "04"  
  
NETWORK 2  
LD      E0.1  
EU  
XMT     VB100, 0       //Start transfer  
  
NETWORK 3  
MEND                    //End main program  
  
//  
//Program start of the SMCI-46 via the serial port  
//Record start for records 1 to 4 via the serial port  
//  
  
NETWORK 1  
LD      SM0.1          //First scan cycle  
MOVB    5, SMB30       //19,200 bauds, 8 data bits, no parity  
  
NETWORK 2 //Program starts  
LD      E0.0  
EU  
MOVB    1, VB100       //1 character will be transferred  
MOVB    16#4, VB101    //Transfer "04"  
XMT     VB100, 0       //Start transfer  
  
NETWORK 3 //Start record 1  
LD      E0.1  
EU  
MOVB    2, VB100       //2 characters will be transferred  
MOVB    16#5, VB101    //Transfer "05"  
MOVB    16#1, VB102    //Transfer "01"  
XMT     VB100, 0       //Start transfer  
  
NETWORK 4 //Start record 2  
LD      E0.2  
EU  
MOVB    2, VB100       //2 characters will be transferred  
MOVB    16#5, VB101    //Transfer "05"  
MOVB    16#2, VB102    //Transfer "02"
```

XMT VB100, 0 //Start transfer

NETWORK 5 //Start record 3

LD E0.3

EU

MOVB 2, VB100 //2 characters will be transferred

MOVB 16#5, VB101 //Transfer "05"

MOVB 16#3, VB102 //Transfer "03"

XMT VB100, 0 //Start transfer

NETWORK 6 //Start record 4

LD E0.4

EU

MOVB 2, VB100 //2 characters will be transferred

MOVB 16#5, VB101 //Transfer "05"

MOVB 16#4, VB102 //Transfer "04"

XMT VB100, 0 //Start transfer

NETWORK 7

MEND //End of main program