MOVITRAC® 31..
Frequency Inverter

Manual
IPOS Positioning Control FPI 31..

Edition 01/97

08/141/93

0922 9469 / 0197

SEW
EURODRIVE

- **Read this manual carefully before you start installation and commissioning work on MOVITRAC® frequency inverters fitted with the IPOS (FPI31..) internal positioning control option.**
  This manual assumes that the user is familiar with and has at his disposal all relevant documentation on the MOVITRAC®31.. system.

- **Safety notes**
  Always follow the safety notes contained in this manual.
  Safety notes are marked as follows:

Electrical hazard, e.g. during live working.

Mechanical hazard, e.g. when working on hoists.

Important instructions for safe and fault-free operation of the driven machine/system, e.g. pre-setting before commissioning.
Failure to follow these instructions may result in injury to people and damage to property.

- **General safety notes for IPOS**
  The IPOS positioning control allows you to match the MOVITRAC® 31.. drive system to the specifics of your application to a very high degree. As with all positioning systems there is, however, the risk of a programming error in the program, which may result in unexpected (not uncontrolled, though!) system behaviour.

- **In these instructions, cross-references are marked with a →, e.g.,**
  (→ MC_SHELL) means: Please refer to the MC_SHELL manual for detailed information or information on how to carry out this instruction.
  (→ Sec. x.x) means: Further information can be found in section x.x of this manual.

- **Each unit is manufactured and tested to current SEW-EURODRIVE technical standards and specifications.**
  The manufacturer reserves the right to make changes to the technical data and designs as well as the user interface herein described, which are in the interest of technical progress.
  A requirement of fault-free operation and fulfilment of any rights to claim under guarantee is that these instructions and notes are followed.
  These instructions contain important information for servicing, they should therefore be kept near the unit.

Contents

1 System Description

The IPOS positioning control provides point-to-point positioning with the MOVITRAC®31.. frequency inverter.

The IPOS positioning control offers the following features:
- Supports fieldbus operation and serial communications through the RS-485 and RS-232 interfaces.
- Relieves the higher-level positioning control (e.g. PLC or IPC) from positioning tasks.
- Economizes on the use of proximity/limit switches compared to positioning by creep speed and use of brake and at the same increases the cycle rate.
- Position-controlled operation also for hoists and when at rest.
- Positioning by tables, a maximum of 32 positions can be permanently stored.
- Teach-in mode.
- High positioning accuracy.
- Programming interface integrated into the MC_SHELL user interface.

Required system components:
- MOVITRAC®31.. frequency inverter, sizes 1, 2, 3 or 4.
- FPI 31.. option pcb, this also provides the encoder supply (5V).
- Incremental encoder (RS-422 / TTL) with zero channel fitted onto the motor shaft. Pulses per revolution: 128, 256, 512, 1024, 2048 ppr, 1024 ppr are preferred.
- Frequency inverter capable of 4-quadrant operation.

Optional expansions (only one option possible at a time):
- Fieldbus interfaces
  Profibus-DP and -FMS with FFP 31.. or Interbus-S with FFI 31..
- Terminal expansion options (in addition to 4/2 digital inputs/outputs in the basic unit)
  – FEA 31.. 4/2 digital inputs/outputs, RS-485, analog functions
  – FIO 31.. 7/6 digital inputs/outputs, RS-485

Setpoint sources:

Setpoint sources positioning control	
Analog input	- Analog input n2 as override input or for teach setpoint
PC interface (USS 11A or UST 11A)	- Control word from PC is combined with terminal functions
FIELDBUS (FFP 31.. or FFI 31..)	- Control and setpoint input by fieldbus possible

Overview of drive structure:
The inverter uses the FPI 31.. to sense the signals from the incremental encoder as the basis for determining speed and drive position. To specify an absolute **reference point** a reference travel must be carried out. For this, 6 selectable strategies **(reference travel types)** are available, which, among other things, determine in which direction of rotation the drive is going to search for the reference point and which of the existing switches (reference cam, CCW/CW limit switch) is used. The limit switches (CCW/CW) determine the limits of travel. The drive is brought to a stop using the rapid stop ramp when approaching a limit switch. Depending on the type of reference travel, a limit switch may also serve as a reversal switch. If the reference cam was not found, the search continues in the opposite direction.

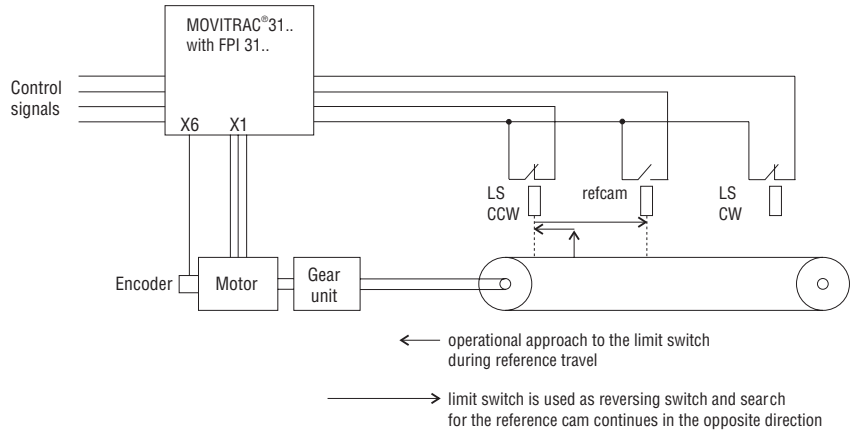


Fig. 1: System overview 00392AEN

The CCW/CW limit switch is defined as the switch which is reached when the motor rotates counterclockwise (n)/clockwise (n0) .

Since the reference cam has a switching hysteresis, the zero pulse of the incremental feedback is evaluated to determine the reference point.

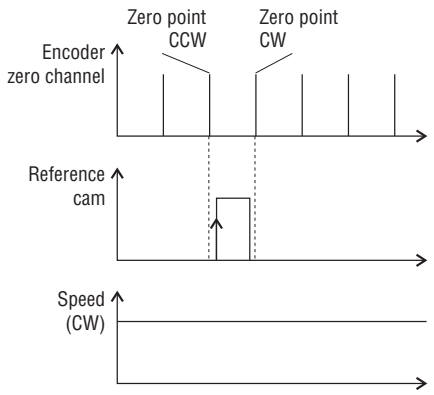


Fig. 2: Zero pulse as reference point 00393AEN

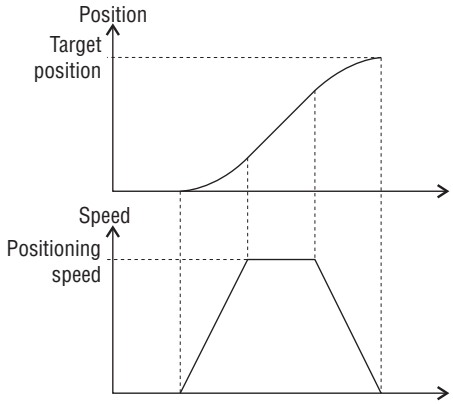


Fig. 3: Speed and position characteristics 00394AEN

Positioning procedure:
When a travel command (GOWA #target position) is entered, a speed profile is generated. The drive then uses this speed profile, which is based on the specified travel parameters (**machine parameters**) positioning speed and positioning ramp, to approach the target position.

Operating modes:

- Automatic mode
  The IPOS program generated by the customer is processed cyclically in the inverter. A PC is required for commissioning and program generation. The drive is controlled by terminal signals or by fieldbus, depending on the degree of expansion. For drive diagnosis the MC_SCOPE user interface is available.
- Manual mode
  The drive can be operated in manual mode from MC_SHELL without a positioning program, e.g. during the initial start up.

A change between modes, i.e. between automatic and manual mode or between IPOS and speed control mode is only possible through MC_SHELL.

2 Installation

2.1 Connecting the cables and the encoder

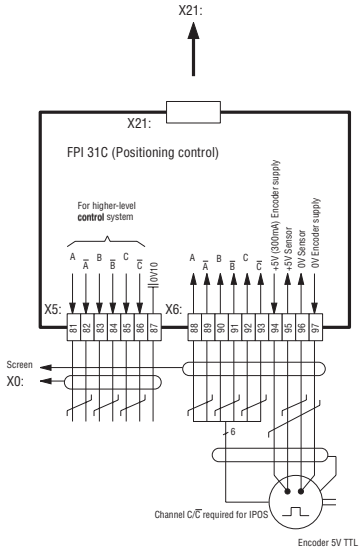


Fig. 4: Wiring diagram for the FPI 31.. 00391AEN

Both the input and output terminals X2/X3 of the basic unit and the input and output terminals X7/X8 of the FEA 31../FIO 31.. option pcbs can be used. The notes in the MOVITRAC® Operating Instructions regarding the encoder connection to FPI 31.. must be observed.

Supported encoder systems are incremental encoders with 128/256/512/1024/2048 pulses per revolution. Only use encoders which are fitted directly onto the motor shaft. If the encoder is not connected to the voltage supply of the FPI 31.., terminals TL 94/95 and TL 96/97 must be jumpered.

2.2 Limit switch connection

All the programmable input terminals can be used as hardware limit switches for the positioning control. The limit switches must be normally closed.

Note: The correct limit switch allocation is always:
- Travel towards “CW limit switch” when motor speed ≥ 0!
- Travel towards “CCW limit switch” when motor speed ≤ 0!

A continuous signal must be ensured in the entire limit switch range!
The purpose of the hardware limit switches is to limit the travel. The range on and outside the limit switches cannot be used for operational travel. During the reference travel, the hardware limit switches act as reversing switches or reference cams, depending on the type of reference travel (→ Sec. 5.2, Reference travel). Approaching the limit switches during a reference travel does not cause a fault.

2.3 Reference cam

Depending on the type of reference travel a reference cam (“normally open”) is required. The reference cam is used as point of reference for determining the drive position.

2.4 Override connection

To enable a travel program to be run at a speed other than its programmed speed, a positive analog setpoint can be entered on terminals X2/34 and X2/35. This allows the traveling speed to be set to a value between 0 and 150% (applied to 0V and 10V) of the programmed speed (e.g. for set-up purposes).
The maximum speed is always limited by the values of parameters P202 (FMAX) and P324 (POLE PAIR NUMBER).

3 Commissioning

3.1 Activating the positioning control

3.1.1 Preliminary work

- Commission the inverter in accordance with the MOVITRAC® 31.. Installation and Operating Instructions in operating mode "V/f control" (parameter P770).
- Commission the inverter in accordance with the MOVITRAC® 31.. Installation and Operating Instructions in operating mode "Speed control" (parameter P770).
- Check the terminal assignment of the inverter when the inverter is in the "NO ENABLE" condition. Program the input/output terminals required for positioning control (→ Sec. 4.3).
  Important: When the operating mode is set to "Positioning" (parameter P770) terminal X2.41 is **automatically** permanently assigned with the function "/controller inhibit" signal.
- Check the limit switches and the emergency stop circuit.
- Set the machine parameters with MC_SHELL (→ MC_SHELL manual).

3.1.2 Setting the positioning control

- To set the operating mode to "Positioning" (P770) the inverter must not be enabled.
- The machine parameters must be set accordingly (see section 4).
- Optimize the speed controller and position controller (see section 3.1.3).
- Develop the position control program.

Operating mode (parameter P770)	IPOS operation
V/f control	Positioning control not possible
Speed control	Positioning control not possible
Positioning	Positioning control is possible if: - no equipment fault (incl. external 24V supply) - enable given (via terminal) / no controller inhibit - direction of rotation enabled - no hold control / no reference travel - limit switches closed (if programmed)

3.1.3 Important parameters for optimization

Summary of the most important parameters which influence the system's control response.

Parameter	Setting*)
P321 Boost	Activating P774 sxR preselection
P322 IxR	
P323 Slip	Motor rated slip
P771 P gain speed controller	For settings see MOVITRAC® 31.. Installation and Operating Instructions
P772 Controller time constant	
P777 P gain feedforward	
Machine parameter gain position controller	Typical values: 0.5 ... 1.5
Machine parameter feedforward	Value < 100 % result in a "soft" run-in to target position

*) Observe setting instructions in the MOVITRAC® 31.. Installation and Operating Instructions.

4 Machine parameters

4.1 General

IPOS machine parameters can only be accessed through MC_SHELL (main menu IPOS), not with the FBG 31.. keypad. The window for setting the machine parameters can be displayed by selecting the IPOS main menu. The most important settings for the positioning control mode can be entered and modified in this window.

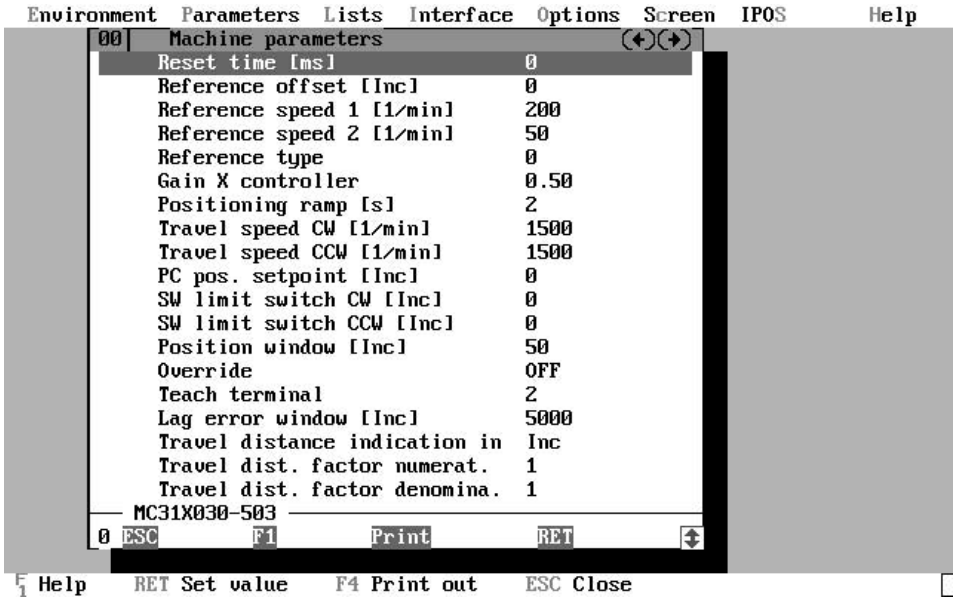


Fig. 5: Machine parameter window 00358AEN

To enable the use of input and output terminals in automatic mode, they must be set accordingly in the "Terminal assignment" menu (→ Sec. 4.3). The machine parameters are "normal" unit parameters and are thus transferred directly to the unit as they are changed and saved accordingly. Like all other parameters, they have a factory-set default value.

4.2 Machine parameter description

4.2.1 Timeout period

The timeout period is the time during which the transfer of data must take place between MOVITRAC® and the PC (or a control system), otherwise the drive will be stopped using the rapid stop ramp. This monitoring function is only active in the IPOS operating mode "Manual".

Range: 0 ... 32767

Factory setting: 0 (monitoring off).

Safety note

Timeout period monitoring ensures that if the link with the PC goes down, the drive stops when the timeout period has elapsed. This is particularly important if a speed or position setpoint is entered manually, but action via the PC is prevented by the failure of the link. It is always possible to switch off via the terminals as well.

4.2.2 Reference offset

This value is added to the reference value to determine the machine zero. The following equation applies:

Machine zero = reference point + reference offset.

As with all length data, this parameter is based on the unit specified by the user (see Sec. 4.2.16).
Factory setting: 0.

4.2.3 Reference speed 1

Determines the speed for the reference travel. The direction of rotation is unambiguously identified by the reference type.
Range: 0 to 5000 rpm.
Factory setting: 200 rpm.

4.2.4 Reference speed 2

Determines the speed for traveling off the reference cam to the encoder zero pulse. The direction of rotation is unambiguously identified by the reference type.
Range: 0 to 5000 rpm.
Factory setting: 50 rpm.

4.2.5 Reference travel type (see section 5.2)

The following are possible:
- **Type 0: no reference travel**
  - machine zero = reference point + reference offset
- **Type 1: CCW end of reference cam**
  - first search direction CCW
  - machine zero = reference point + reference offset
- **Type 2: CW end of reference cam**
  - first search direction CW
  - machine zero = reference point + reference offset
- **Type 3: CW limit switch**
  - first search direction CW
  - machine zero = reference point + reference offset
- **Type 4: CCW limit switch**
  - first search direction CCW
  - machine zero = reference point + reference offset
- **Type 5: no reference travel**
  - machine zero = reference offset

Note: This parameter must not be changed whilst a reference travel is being performed (fault 61)!
Factory setting: type 0.

4.2.6 Position controller gain

Value for the P controller of the position control loop. The value can be optimized with MC_SCOPE in conjunction with the speed controller.
Range: 0.1 to 60.
Factory setting: 0.5.

4.2.7 Positioning ramp

Value for the ramp time during the positioning process. The standard ramps (P12_, P13_) are not effective. The **same** ramp is used for both acceleration and deceleration. The value represents a time to ramp the drive from 9 to 3000 rpm.
Range: 0 to 0.5 s → 0.02; 0.5 to 3 s → 0.1; 3 to 10 s → 0.5.
Factory setting: 2 s.

4.2.8 Positioning speed CW/CCW

These two parameters determine the positioning speed in the positive direction (n > 0) / in the negative direction (n < 0).
The value is entered in revolutions per minute and must be matched to the maximum motor speed.
Range: 0 to 5000 rpm**.**
Factory setting: 1500 rpm.
Note: Parameters 202 (pole pair number) and 324 (f_{max}) limit these machine parameters (max. 120 Hz)!

$$n_{max} = \frac{P202 \bullet 60}{P324}$$

4.2.9 PC position setpoint

Can be used at any point within the automatic program with the **GOPA** command. The value is entered in MC_SHELL in user units and via the communications interfaces in increments. The parameter value is not stored when the unit is powered down.
Range: -2^{31} to $+2^{31}$ increments.
Factory setting: 0.

4.2.10 Software limit switches CW/CCW

These two parameters determine the limits of travel. In the automatic mode they ensure that the drive does not go to a target position outside the limits of travel. If the target position lies outside the limits of travel, the drive immediately decelerates using the rapid stop ramp and generates a fault message. To clear a fault after approaching a software limit switch refer to section 5.5. In the case of reference travel types 3 and 4 the corresponding software limit switch must be set to the position value of the hardware limit switch used as reference cam. For uni-directional operation both values must be set to 0. The value is entered in MC_SHELL in user units and via the communications interfaces in increments .
Range: -2^{31} to $+2^{31}$ increments.
Factory setting: 0 (Monitoring of the software limit switches deactivated).
Note: Monitoring of the software limit switches is only effective after a reference travel.

4.2.11 Position window

This parameter is used to indicate when the target position has been reached. This is done by continuously comparing the absolute difference between actual position and target position. If this difference is less than the value specified for the position window then the axis is considered "In position", and an output assigned with the function "Axis in position" is set high.
Range: 0 to 2^{15} increments.
Factory setting: 50.
Note: If the brake function is activated, the position window value must be greater than the brake tolerance.

4.2.12 Override

Switches the override function ON and OFF (see Override connection, Sec. 2.4).

4.2.13 Teach terminal

This parameter determines which of the physical input terminals (basic unit or FEA 31../FIO 31..) is used for the teach function. The physical input terminals are numbered consecutively.

Terminal strip	**X3** Basic unit			**X8** FEA 31../FIO 31.. option				**X19** FIO 31.. option		
Terminals	42	43	47	48	49	50	51	52	53	54
Teach terminal value	1	2	3	4	5	6	7	8	9	10

Range: 1 to 10.
Factory setting: 2.

4.2.14 Lag error window

The lag error window is the maximum permissible difference between target position and actual positions. If this is exceeded, a lag error occurs. The motor is brought to a stop using the rapid stop ramp. Lag error monitoring is deactivated if this parameter is set to 0.
Unit: increments.
Range: 0 to 2^{31} increments.
Factory setting: 5000.

4.2.15 Travel distance indication

Five-digit character string for the user-defined length units. These are freely selectable and are stored in the unit. In the manual mode window, the machine parameter window and in the positioning program, all length entries will be shown with this symbolic description.
Note: This parameter has no travel-specific effect.

4.2.16 Travel distance factor numerator/denominator

These parameters are used to convert user length units into incrementals. IPOS always operates internally with 4096 incr./revolution, regardless of the number of pulses of the incremental encoder. The general conversion equation is:

$$x_{IPOS}\ [\text{increments}] = \frac{\text{travel distance factor numerator}}{\text{travel distance factor denominator}} \bullet x_{command}\ [\text{travel distance indication}]$$

The numerator represents the number of increments per output shaft revolution, the denominator represents the traveled distance per revolution in user units.
Range: 0 to 2^{31}.
Factory setting: 1.

Example: The drive shown below (bottling application) is to be programmed in the following units:
a) **mm** in the linear axis
b) **increments**
c) **output revolutions**
d) **bottles** (3 bottles / 400 mm)

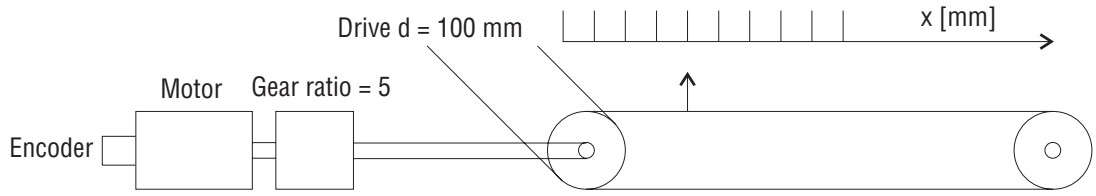


Fig. 6: Example for programming units 00395AEN

Solutions:

a) User distance units = **mm**
Travel factor numerator = (4096 · gear ratio i) · reduction factor = 4096 · 5 · 1000 = 20480000
Travel factor denominator = (d_{output} · π) · reduction factor = 100 · π · 1000 = 314159

Note: The conversion can be made more accurate if the numerator and denominator are multiplied by a factor to eliminate the decimal portion (only required if numerator or denominator are not integers). This does **not** limit the maximum travel range. The travel distance factors are stored in the axis, the conversion is carried out by the PC operator routine.

b) User distance units = **incr.**
Travel distance factor numerator = 1
Travel distance factor denominator = 1

c) User distance units = **rev.**
Travel distance factor numerator = 4096 · gear ratio = 4096 · 5 = 20480
Travel distance factor denominator = 1

d) User distance units = **bot.**
Travel distance factor numerator = (4096 · gear ratio i) · 400(mm) · reduction factor
= 4096 · 5 · 1000 = 8192000000
Travel distance factor denominator = d · π · 3 (bot) · reduction factor = 100 · π · 3 · 1000
= 942477

4.2.17 Brake function

This function supports control of a mechanical motor brake in positioning mode. If the brake function is activated, the brake reaction time is taken into account when stopping and starting. The brake output X3.61 and the output stage are then controlled appropriately.

Note: The brake function can be switched on and off by a command in the automatic program (→ Sec. 7.6.8). If the brake function is used, the position window must be larger than the tolerance of the mechanical brake (to take the backlash of the brake carrier and the brake pad into account).

Response of the axis with braking function on:

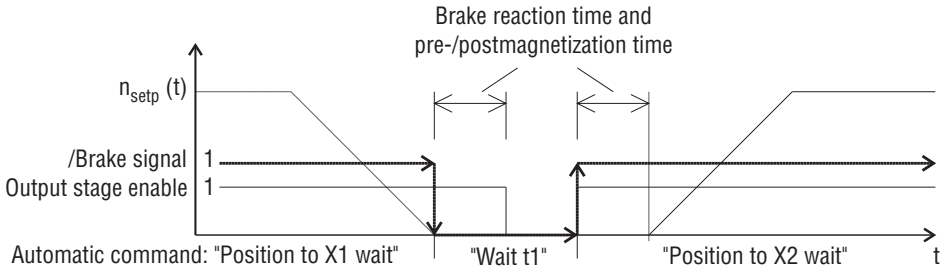


Fig. 7: Brake function 00396AEN

The brake function is always effective if IPOS is being operated in position controller mode (and the function is activated). This will be the case in automatic mode and manual mode/position control.
The brake will always be released in manual mode/speed control!

4.2.18 Brake reaction time

This parameter allows for programming of the brake reaction time of the mechanical brake.

4.2.19 IPOS bus mode

This parameter enables IPOS to be controlled via a fieldbus interface and determines how a position setpoint transmitted via fieldbus is evaluated.
The following settings are possible:

IPOS bus mode	Use of the fieldbus setpoint position
0	Bus position setpoint not used
1	The bus position setpoint is used as manual mode setpoint
2	The bus position setpoint is used for the GOPA command in the automatic program
3	Reserved

In IPOS bus mode = 2 → the position setpoint is stored in variable 254 and can be further processed in the program.
Range: 0 to 3.
Factory setting: 0.

4.2.20 Feedforward

With a value of 100%, the drive operates with a linear speed profile optimized to respect of time, minimizing the difference between setpoint speed and actual speed. If a value less than 100% is specified, a gap between setpoint position and actual position occurs (lag distance) during a positioning operation. This results in a "soft" run-in to the target position.

Setting instructions:

Desired control response	Feedforward setting	Advantage	Disadvantage
Time-optimal positioning with minimum lag distance	100 %	– minimum travel time – short lag distance	– block-type torque characteristic – "hard" run-in to target position
"Soft" run-in to target position	50% to 80%	– "soft" run-in to target position	– greater lag distance – positioning process takes longer

Range: -150% to +150%.
Factory setting: 100%.
The lower the value is set, the sooner it trips the axis with a lag error (F42). The lowest possible setting depends on the required dynamics of the system.

4.3 Basic unit parameters important for IPOS

Menu item 770: operating mode

Set to ***Positioning***.

Menu item 140: rapid stop ramp

This ramp is used in conjunction with the **STOP** command or if the enable signal is removed. It is further used during the entire reference travel and for fault response.

Menu item 777: feedforward gain and 778: feedforward filter

The setting is the same as for the speed control (→ MOVITRAC® 31.. Installation and Operating Instructions).

Menu item 324: pole pair number and 202: F_{max} (max. 120 Hz)

These values limit the machine parameters "Positioning speed CW/CCW". The speed value resulting from these parameters should always be larger than the desired positioning speed (otherwise a lag error may occur).

$$f_{max} = \frac{P202 \bullet 60}{P324}$$

Menu item 323: slip

This parameter should be set based on general commissioning data.

Menu items 600 to 606: binary inputs on basic unit and FEA 31../FIO 31..

The input terminals initially have the functions and characteristics as described in the MOVITRAC® 31.. Installation and Operating Instructions. In addition, IPOS allows the functions "/limit switch CW", "/Limit switch CCW", "Reference cam" and "Reference travel" to be programmed to the input terminals. Input terminals, which are used in the IPOS automatic program **exclusively** must be programmed with "No function". To use them in the automatic program, a "1" must be entered in the respective positions in the selection masks for the individual commands.

The following IPOS commands are possible:

- wait until terminals are 1 or 0,
- jump if terminals are 1 or 0,
- selection of table positions with terminals,
- teach function,
- setting terminals

Menu items 611 to 613: binary outputs on basic unit and FEA 31../FIO 31..

Functions that are used in conjunction with IPOS:

- axis in position,
- 8 freely programmable binary outputs of which 3 can be physically implemented with FEA 31.. and
  7 with FIO 31...

Output terminals, which are used in the IPOS automatic program exclusively must be programmed with "Output IPOS x". (x = 1 ... 8).

5 Working with IPOS

5.1 IPOS operating modes

The following operating modes are available in IPOS:

- reference travel (see Sec. 5.2)
- manual mode (see Sec. 5.3)
  - manual mode speed control
  - manual mode position control
- automatic mode (see Sec. 5.4)
  - automatic STOP
  - automatic RUN
  - automatic STEP
  - automatic BREAKPOINT
  - automatic HALT

5.2 Carry out reference travel

Referencing provides a way of determining a machine zero to which all positioning commands refer. To ensure that positioning operations remain constant over many years it must be possible to define this point most precisely and with a high repeat accuracy.

Since the reference cam has a switching hysteresis, the nearest **zero pulse** of the encoder system is used as a **reference point**. Whether this is the CW or the CCW one is determined by the reference travel type. Once the reference point is known, the machine zero (actual reference point for all positioning commands) can be determined based on the reference point. The following applies:

Machine zero = reference point + reference offset

The reference cam must be normally open. The first search direction is determined by the type of reference travel. During reference travel, the limit switches act as “reversing switches”. When the reference travel is completed, the drive remains position controlled in the position “machine zero”. The rapid stop ramp is used during the entire reference travel (necessary to ensure rapid change of the various reference speeds).

Note:
The reference cam must be wide enough to ensure unambiguous behaviour for the appropriate reference travel types and the specified reference speeds together with the rapid stop ramp (important for types 1 and 2).

5.2.1 Types of reference travel

Explanation of the reference travel type diagrams

Possible starting points for the drive:

① drive is located between reference cam and CW limit switch
② drive is located on the reference cam
③ drive is located between reference cam and CCW limit switch

Explanation of the abbreviations in the diagrams:

CW LS	CW hardware limit switch (positive limit switch)
CCW LS	CCW hardware limit switch (negative limit switch)
ref cam	reference cam
NP	zero pulse ≙ reference point
Vref1	initial reference travel speed
Vref2	final reference travel speed to approach the zero pulse of the encoder.

Type 0: Reference point is the first zero pulse CCW to the current position.
machine zero = zero pulse CCW to rotor position + reference offset

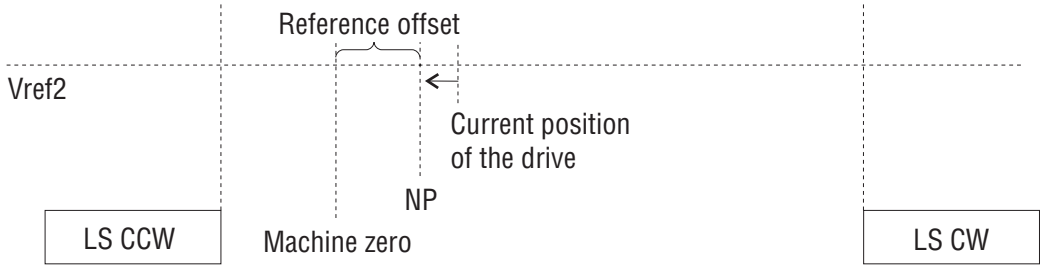


Fig. 8: Reference travel type 0

00397AEN

Note: Only when the command GOWA (#0) is given, does the drive travel to the machine zero (v_{ref2}).

Type 1: Reference point is the first zero pulse CCW to the reference cam
machine zero = reference point (NP) + reference offset

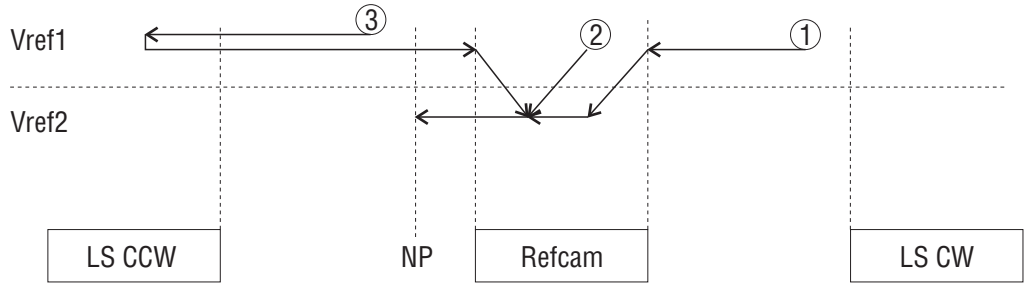


Fig. 9: reference travel type 1

MD0066AE

Type 2: Reference point is the first zero pulse CW to the reference cam
machine zero = reference point (NP) + reference offset

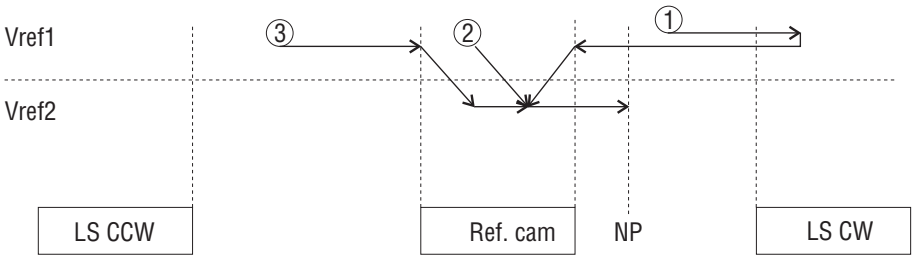


Fig. 10: Reference travel type 2 MD0067AE

Type 3: Reference point is the first zero pulse CCW to the CW limit switch
machine zero = reference point (NP) + reference offset
Reference cam is not required.

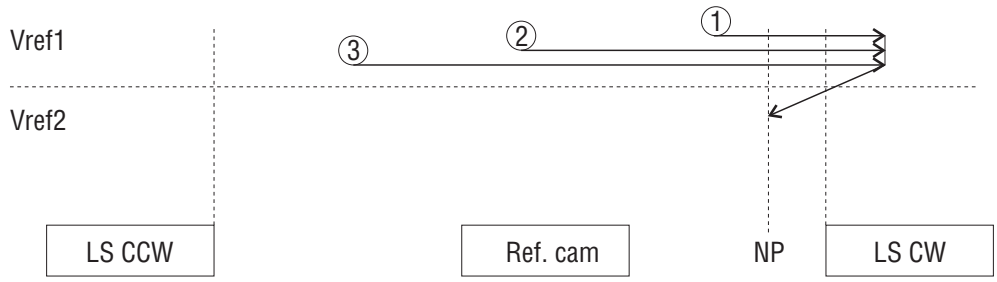


Fig. 11: Reference travel type 3 00398AEN

Type 4: Reference point is the first zero pulse CW to the CCW limit switch
machine zero = reference point (NP) + reference offset
Reference cam is not required.

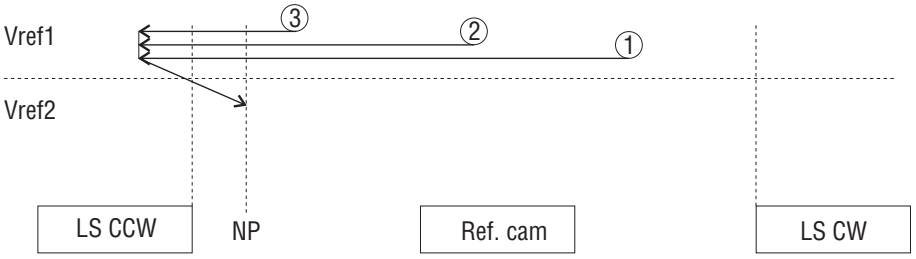


Fig. 12: Reference travel type 4 00399AEN

Type 5: No reference travel: machine zero = reference offset
Note: Reference travel must also be initiated for reference travel type 5.

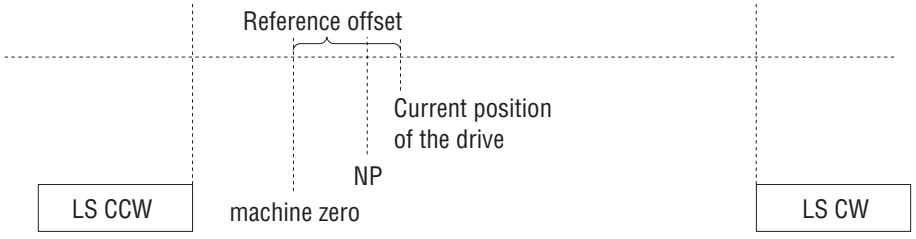


Fig. 13: Reference travel type 5 00018AEN

Reference travel can be initiated in the following ways:

- By programming an input terminal (basic unit or FEA 31../FIO 31..) with the "REFERENCE TRAVEL" function. When the signal goes high, the reference travel is performed according to the selected reference travel type.
- By a **GOØ** command in the automatic program.
- By making an entry in the manual mode window.

Once started, a reference travel will be completed even if the initiating signal is no longer there. The reference travel procedure is interrupted when the enable signal goes low and cancelled if the inhibit signal goes low. An axis that is already referenced can only be referenced again from the manual mode window or with the GOØ #1 command in the automatic program. Completion of the reference travel is indicated in the status window.

5.3 Manual mode

Manual mode can be initiated by opening the manual mode window (IPOS main menu). The operating mode (parameter P770) must be set to "Positioning".

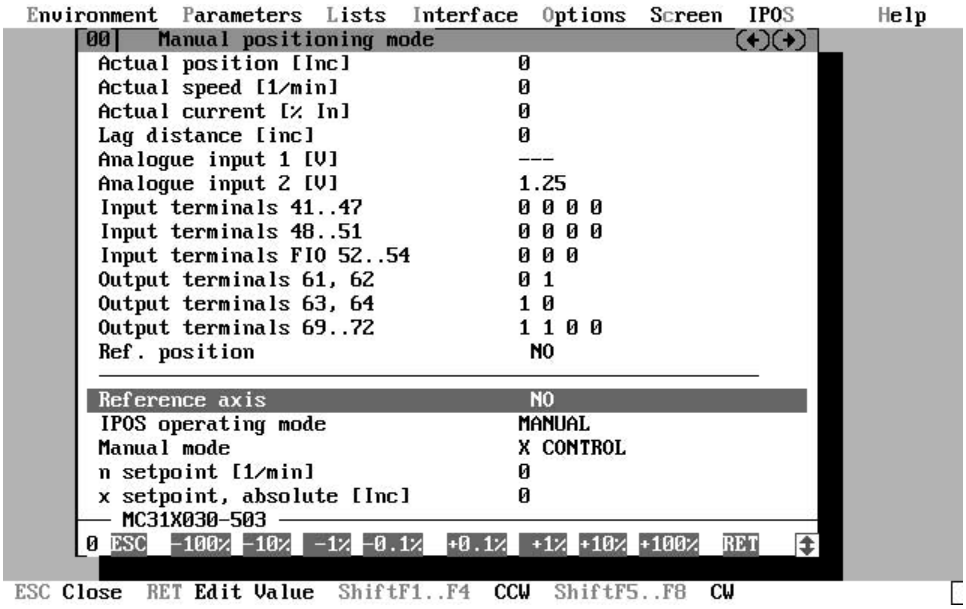


Fig. 14: Manual mode window 00359AEN

The upper part of the manual mode window provides status information on the axis. The lower part contains the control features used for manual operation of the axis.

A **TEACH FUNCTION** allows for the argument (distance) of a GO-command in the automatic program to be set to the position which has been approached in manual mode . To do this, highlight the travel command in the automatic program with the cursor bar. Click on the TEACH button or press the **ALT + F9** keys to get to the manual mode window. You can either "teach" the actual position or travel to the required postion. Press the **ESC** key and then confirm with **RETURN** to set the highlighted travel command with this position.

5.3.1 Reference axis

The axis carries out a reference travel according to the selected machine parameters. A referenced axis can be referenced again with this menu item. If a value not equal to zero is entered in "position setpoint absolute", the drive will travel to this position upon completion of the reference travel. When the referece travel is completed, this is indicated in the status section of this window.

5.3.2 IPOS operating mode

Select “Manual mode” to operate the drive manually.

5.3.3 Manual mode

Specifies whether the axis is operated in speed control (n control) or position control (x control) mode. According to the setting, the n-setpoint or the x-setpoint will be used. The positioning ramp (machine parameters) is used in both settings. If the drive is operated in position control, the specified traveling speeds are used.

5.3.4 Speed setpoint

This value is used as a speed setpoint if ***manual mode*** “speed control” is selected. The value is activated and weighted by operating the buttons -100% to +100%. If the buttons are not used, the default setpoint = 0 is used.

5.3.5 Absolute position setpoint

The value is entered in user distance units (Machine parameters → Sec. 4) and used as an absolute target position if confirmed with the ENTER key. The selected travel parameters will be used.

5.3.6 Relative position setpoint

The value is entered in user distance units (→ Machine parameters, Sec. 4) and added to the current position value if it is confirmed with the ENTER key. The selected travel parameters will be used.

5.3.7 Travel parameters

The parameters ***travel speed CW, travel speed CCW, positioning ramp*** are described in the machine parameters section (→ Sec. 4).

5.3.8 Timeout period

If the serial link goes down, the axis executes a rapid stop after the specified timeout period has elapsed. A timeout does not cause a unit trip.

Note: If a value of 0 is entered, timeout monitoring is switched OFF. In this case the setpoint transmission is not monitored and an interruption of the PC-MOVITRAC® 31.. serial link is not recognized.

Safety note

Timeout period monitoring ensures that the drive stops after this period if the PC link goes down. This is of particular importance if a speed or position setpoint is entered in manual mode, but intervention from the PC is not possible as a result of the link going down. It is in any case always possible to switch off via the terminals.

5.4 Automatic mode

Automatic mode consists of: STOP, START, STEP, BREAKPOINT, HALT. The program can be entered and viewed in the **Positioning** window. The status section of the **Positioning** window shows the current automatic mode settings. The settings are activated with the function keys [F4/F5/F7/F8/F9].

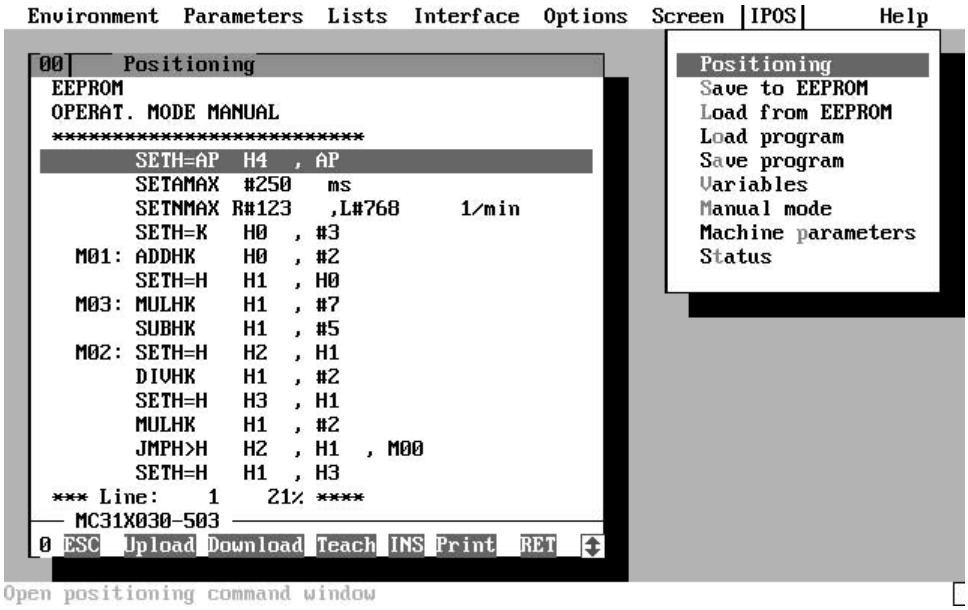


Fig. 15: Positioning window 00360AEN

5.4.1 Automatic BREAKPOINT [F4]

Program execution continues until the specified break address (cursor bar) is reached. The program line highlighted with the cursor bar is not executed. If the break address cannot be reached, the automatic program runs continuously.

5.4.2 Automatic HALT [F5]

The drive is brought to a stop (rapid stop ramp) and the automatic program is reset. After the axis has come to rest, the unit switches to automatic STOP and can then be restarted.

5.4.3 Automatic STEP [F7]

Only the current command in the automatic program is executed. The arrow in front of the command line indicates the line which will be executed next. The axis changes automatically into automatic mode STOP after one command has been executed and the axis is at rest.

5.4.4 Automatic STOP [F8]

The automatic program is stopped at the current position. Position control or any active travel command will be continued and completed. The current line number of the IPOS program will be indicated. The stopped program can be resumed by pressing the [**F9]** function key.

5.4.5 Automatic START [F9]

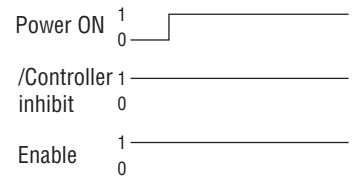
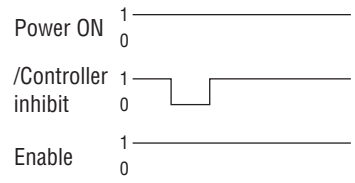
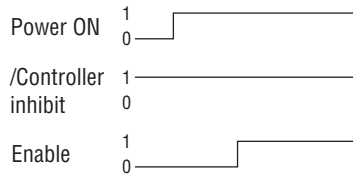
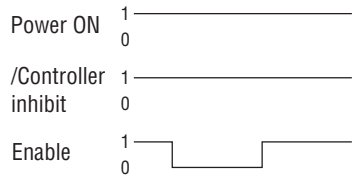
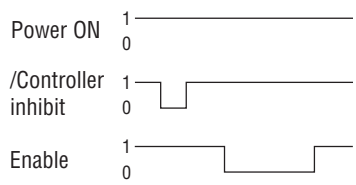
The automatic program is started and runs continuously (looping program). The current program line number is shown in the **Positioning** window.

Note: To start execution of the program with no PC connected the START mode must be active when exiting MC_SHELL.

5.4.6 Controller inhibit and enable function for IPOS

Controller inhibit and rapid stop (no enable) interrupt the IPOS processing (automatic or manual mode) regardless of whether the drive is operated with the physical unit terminals, from the MC_SHELL panel or via a fieldbus interface. In operating mode "Positioning" (P770) the "/controller inhibit" function is fixed assigned to TL 41 of the basic unit. The default setting for TL 43 is "enable"; it may however be routed to any other programmable terminal. The following IPOS functions can be implemented with the controller inhibit and rapid stop functions:

- Reset automatic program without PC
- Reset an IPOS fault without PC
- Restart the automatic program without PC
- Stop the drive axis and continue the automatic program at the point of interruption.
- Lift the limit switch lockout to allow travel clear of the hardware limit switches (→ Sec. 5.5).

Signal patterns of: PowerON: /Controller inhibit: Enable:	**Last setting of IPOS operating mode from MD_SHELL user interface**	
	Automatic RUN	Manual mode
	The IPOS automatic program starts with the first IPOS command. No "start terminal" (enable) is necessary.	IPOS manual mode with setpoint = 0 until new setpoint is sent from controller or MC_SHELL.
	Automatic program is reset, operating mode HALT (after axis has stopped → STOP mode). Automatic program does not start.	Automatic program is reset, operating mode HALT (after axis has stopped - STOP mode).
	Automatic program starts after enable. Automatic program is not reset.	IPOS manual mode with setpoint = 0 until new setpoint is sent from controller or MC_SHELL.
	The automatic program is interrupted by "No enable" and then continues from the point of interruption.	Manual mode is interrupted by "No enable" and then continues from the point of interruption.
	The automatic program is interrupted and reset by controller inhibit. The automatic program is started after the signal level on the enable terminal changes from 0-1. IPOS faults are reset.	Manual mode is interrupted by controller inhibit. Manual operation is set again after the signal level on the enable terminal changes from 0-1. IPOS faults are reset.

5.5 Limit switch processing

5.5.1 Hardware limit switches

If IPOS senses a hardware limit switch, the drive will be brought to a stop using the rapid stop ramp and then remains on the limit switch (display hardware limit switches CW/CCW). The output stage is turned off and if the motor has a brake, it is then set.

To travel clear off the hardware limit switch at reference speed 2, reset the terminal "controller inhibit" $1 \rightarrow 0 \rightarrow 1$. The drive stops as soon as the limit switch terminal signal turns high again (limit switch cleared). If no PC is connected a $1 \rightarrow 0 \rightarrow 1$ switching sequence for the Enable signal is required to restart the program.

Exception:

- The drive is carrying out a reference travel, the limit switches are then used as reversing switches according to the selected reference travel method (see description of reference travel).

The limit switch monitoring (limit switches reversed, limit switch open circuit) is also effective if the limit switches are sensed in the IPOS mode.

Note: Note that the drive is brought to a complete stop **while on the limit switch**. If the drive passes the limit switch, traveling clear off the limit switch is not possible in automatic mode. The unit trips with F29 (hardware switches swapped around) when trying to travel back into the permissible travel range.

5.5.2 Software limit switches

If the calculated target position lies outside the permissible travel range (software limit switch range) the drive is immediately brought to a stop, regardless of if the absolute position of the drive has exceeded the setting of the software limit switches or not.

If the actual position lies within the software limit switch range, fault F59 can be cleared with a switching sequence X2.41 = $1 \rightarrow 0 \rightarrow 1$ and then X3.43 (enable) = $1 \rightarrow 0 \rightarrow 1$. If the drive has left the software limt switch range, the value of the software limit switches has to be set to zero or a reference travel has to be initiated.

6 Fault messages / Service information

6.1 Status displays

6.1.1 Status window

All important information about the state of the connected inverter is shown in the “IPOS” status window.
In detail, these are:
– Fault indication
– Actual position in user distances units (see Machine parameters)
– Actual speed value in rpm (see Machine parameters)
– Actual current
– Lag distance (in increments)
– Analog voltage 1
– Analog voltage 2
– Basic unit binary inputs
– FEA 31../FIO 31.. binary inputs
– Basic unit binary outputs
– FEA 31../FIO 31.. binary outputs
– Indication whether the axis is referenced or not

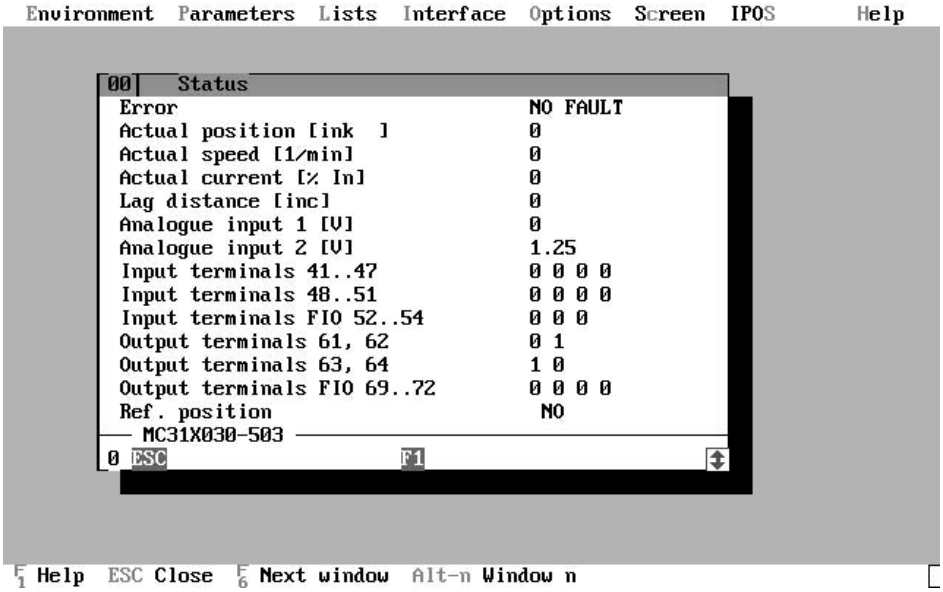


Fig. 16: Status window 00362AEN

6.1.2 Display window “Process value”

The “Process value” window (main menu “Parameters”, menu item “Display values”) provides further information important for operating the MOVITRAC® 31.. with IPOS (see MOVITRAC® 31.. Operating Instructions).

6.2 IPOS fault response

6.2.1 List of IPOS faults

The following list provides an overview of all fault messages relevant to IPOS operation.

Fault number	Meaning	Possible causes	Remedy	Read y message	/F ault message	Response
50	Limit switch missing	Limit switch not connected or wire break	Check the limit switch wiring		X	*
51	Zero pulse missing	Encoder defective or cable break or C/C channel (zero channel) of the encoder not connected to FPI 31..	Check the wiring of the encoder; check the encoder for correct functioning		X	*
52	Limit switches reversed	Incorrect wiring	Check the wiring of the limit switches		X	*
55	Invalid IPOS command	No program or wrong program (e.g. after factory setting)	Check the content of the program memory	X	X	Rapid stop
56	User watchdog (in positioning program)	System fault or wrong time setting	Check use of the watchdog function	X	X	Rapid stop
57	Teach error	Teach procedure incorrect	Check teach procedure	X	X	Rapid stop
58	Invalid control word	An attempt has been made to set an invalid automatic mode via serial link or fieldbus.	Check the serial connection and the setting value of the external control	X	X	Rapid stop
59	Software limit switches	Target position lies outside the software limit switch range	Check the software limit switch range and the travel program	X	X	Rapid stop
60	Lag error	System fault or lag error window too small or controller setting not optimized	Check setting of the parameter lag error window or optimize controller setting	X	X	Rapid stop
61	Fault during reference travel	Reference cam not present or incorrect limit switch connection or reference type changed during reference travel	Check the reference type and its corresponding requirements		X	*
62	Index overflow	Basic programming rules not observed (6.1.1), causing stack overflow	Check and correct user program		X	*
63	Jump command error	Jump to invalid label	Check the positioning program	X	X	Rapid stop
64	Limit switch CW approached	Limit switch CW approached or not connected or wire break	Check the travel program or the limit switch wiring	X	X	Rapid stop
65	Limit switch CCW approached	Limit switch CCW approached or not connected or wire break	Check the travel program or the limit switch wiring	X	X	Rapid stop

* Rapid stop with unit shutdown / XX = LED V1 flashing red/green / further fault numbers → MOVITRAC® 31.. Installation and Operating Instructions

6.2.2 Fault reset

Faults which do not cause the unit to switch off, enable the drive to continue in operation without a new reference travel. After eliminating the actual cause of the fault, the following methods may be used to reset an IPOS fault:

1) Controller inhibit / enable, see Sec. 5.4.6
2) By setting one of the following functions in the MC_SHELL user interface:
   - AUTOMATIC START
   - AUTOMATIC BREAKPOINT
   - AUTOMATIC STEP
3) Selecting parameter **862** KEYPAD RESET
4) By switching the power OFF/ON.

When a fault occurs, the axis goes into automatic HALT mode (automatic program is reset, drive stops). After the drive stops, automatic STOP is set. Reset of an IPOS fault activates the previous operation mode (active before the fault occured). An IPOS fault is shown by the **fault** output terminal function.

7 Travel programs

7.1 Programming travel programs

Program entry is done as described in Sec. 7.4 **Entering programs**. It is not possible to enter and create IPOS programs with a text editor outside MC_SHELL.

7.1.1 General IPOS programming rules

IPOS provides commands that support a structured application solution using an automatic program. These are:
- Subroutine system
- Loop command with begin and end of loop (block structure) and specification of number of loop cycles (nested loops possible)
- Use of labels as jump destinations with automatic calculation of the absolute address.

Only approx. **100** program steps can be programmed. **256** variables are available, of which **32** can be stored to non-volatile memory. When the main program has ended, the program starts over out the first line (copping program). The reference travel is only initiated by the **GOØ#0** command if the axis has not been referenced yet.

Safety note: Subroutines must never be exited with a jump command. If a subroutine is to be exited conditionally, jump to the end of the subroutine. A LOOP block must never be exited with a jump command. Jumps within a LOOP block are permissible. Failure to comply with these notes will result in program execution errors.

7.2 Program format

Syntax:	**Explanation:**
command	No specific program header is required
[comment]	Comments are only saved to the file
CALL [Label 1]	Call first subroutine
RET command	End of main program
[Label 1] command	Start of first subroutine (via Label)
[comment]	
RET command	End of subroutine
END command	End of complete program

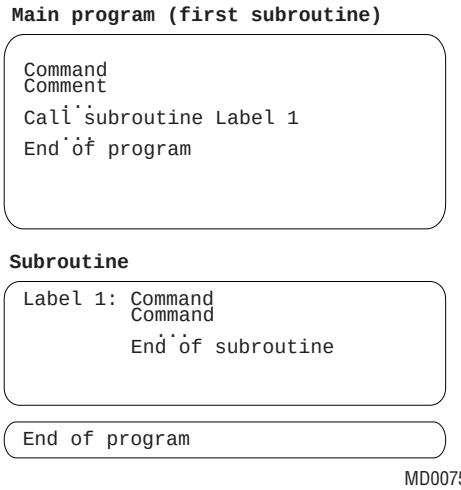


Fig. 17: Block diagram program format

Example:

```
Main program
    GOØ    #Ø
    GOWA   #1Ø     rev.
    CALL   MØ1
    GOWA   #25     rev.
    RET

Terminal toggle subroutine
    MØ1 :  SETO1   #0000 0001
    WAIT   #1ØØØ   ms
    SETOØ  #0000 0001
    WAIT   #1ØØØ   ms
    RET
    END
```

The main program first carries out a reference travel, positions absolutely to +10 revolutions and then calls the subroutine. In the subroutine, the first IPOS output terminal is pulsed once. After the subroutine has ended, the program executes a travel to position + 25 rev. The program returns to the top and runs continuously.

7.3 Loading/saving/editing programs

If you want to enter a new IPOS program or adapt existing programs, select the **Positioning** menu under the ***IPOS*** main menu. When the window is opened with an inverter connected, the positioning program stored in the inverter is loaded. To save a program, select the ***Save program*** menu item. This allows a program in the positioning window to be saved as a file when a file name is entered. To name a new program press the "right arrow" key or select "Name:" with the mouse. All program files must have a ".prg" extension. The menu item ***Load program*** is used to load a positioning program from a file into the **Positioning** window.

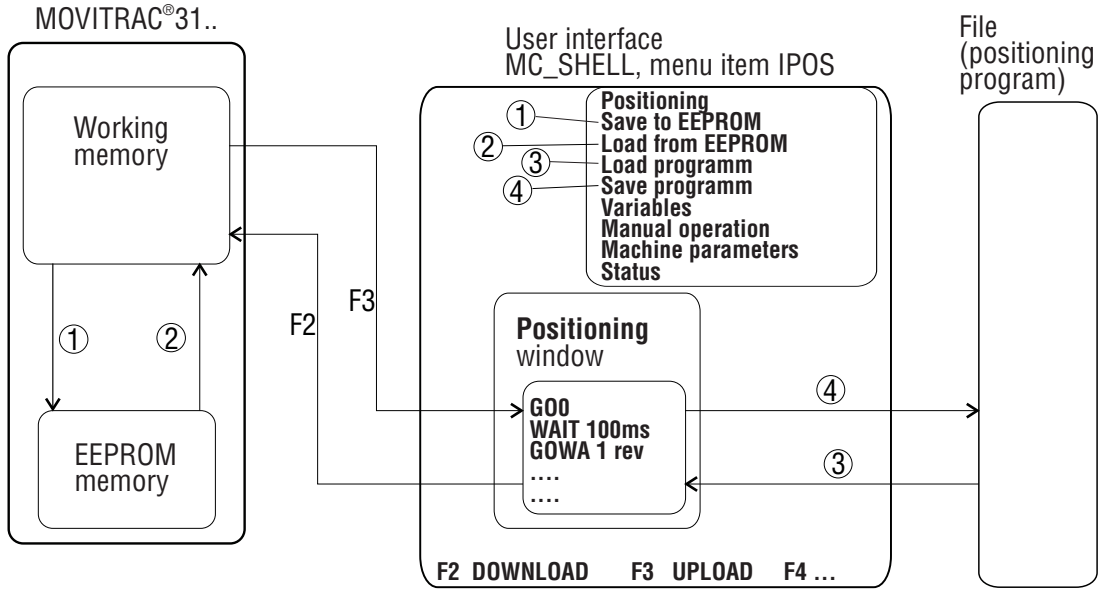


Fig. 18: Uploading positioning programs

00355AEN

Positioning programs are uploaded into the positioning window from an inverter with F3 **UPLOAD**. In the opposite way, F2 **DOWNLOAD** transfers a positioning program to the inverter from the positioning window (only possible if X2.41 = 0, controller inhibit off). The program is thus stored in the IPOS user RAM. To store it permanently in the unit (i.e. even with power off), the **Save to EEPROM** menu item must be used. During this process an "X" appears in the status section of the Positioning window. This procedure takes about 15 s and can take place parallel to program execution.

Note: When downloading a program to MOVITRAC® 31.. memory, all comment lines are lost. Programs with comment lines can only be saved as files on the PC.

7.4 Entering programs

Positioning programs are written or edited in the ***positioning*** window of the user interface. When the "Positioning" window is open, the command selection table can be accessed by pressing the INSERT key. The command selection table lists all the IPOS commands. By selecting the desired command and confirming with ENTER, a mask for the entry of command arguments is opened. The desired values should be entered here. If the entries are confirmed with ENTER, the command will be transferred to the "Positioning" window. F1 provides a help text for each command.

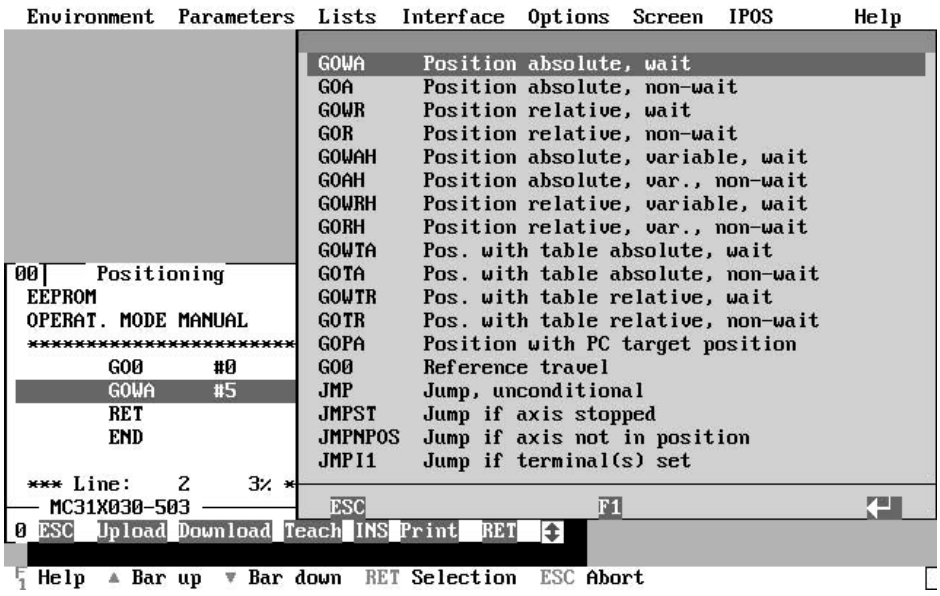


Fig. 19: Editing of positioning programs

00361AEN

It is also possible to copy program lines within the "Positioning" window. For details refer to the main menu "Help" (menu item "key assignment") in MC_SHELL.

7.5 Set of commands

7.5.1 Overview

Positioning commands:	
GOWA	Position absolute, wait
GOA	Position absolute, non-wait
GOWR	Position relative, wait
GOR	Position relative, non-wait
GOWAH	Position absolute to variable, wait (only in increments)
GOAH	Position absolute to variable, non-wait (only in increments)
GOWRH	Position relative to variable, wait (only in increments)
GORH	Position relative to variable, non-wait (only in increments)
GOWTA	Position absolute to table position, wait
GOTA	Position absolute to table position, non-wait
GOWTR	Position relative to table position, wait
GOTR	Position relative to table position, non-wait
GOPA	Position absolute to PC setpoint, non-wait
GO ∅	Carry out reference travel
Jump commands:	
JMP	Unconditional jump to label
JMPST	Jump to label if axis has stopped
JMPNPOS	Jump to label if axis not in position
JMPI1	Jump to label if input terminal(s) = 1
JMPI ∅	Jump to label if input terminal(s) = ∅
JMPT ∅ = ∅	Jump to label if timer ∅ = ∅
JMPT1= ∅	Jump to label if timer1 = ∅
JMPH >K	Jump to label if variable greater than constant
JMPH <K	Jump to label if variable less than constant
JMPH =K	Jump to label if variable equal to variable
JMPH >H	Jump to label if variable greater than variable
JMPH <H	Jump to label if variable less than variable
JMPH =K	Jump to label if variable equal to variable
JMPAP >K	Jump to label if actual position greater than constant
JMPAP <K	Jump to label if actual position less than constant
JMPAP >H	Jump to label if actual position greater than variable
JMPAP <H	Jump to label if actual position less than variable
JMPCUR >K	Not available for IPOS operation with MC 31...
JMPCUR <K	
JMPCUR >H	
JMPCUR <H	
Wait commandds:	
WAIT time direct	Wait time direct
WAITI1	Wait until input terminal(s) = 1
WAITI ∅	Wait until input terminal(s) = ∅
WAITPOS	Wait until drive in position

Set commands:	
SETO1	Set output terminal(s) = 1
SETO ∅	Set output terminal(s) = ∅
SETNMAX	Set travel speed
SETNMAX=H	Set maximum travel speed with variable
SETMMAX	Not available for IPOS operation with MC 31...
SETMMAX=H	
SETAMAX	Set positioning ramp
SETAMAX=H	Set positioning ramp with variable
SETT ∅	Set timer ∅
SETT1Set timer 1	Set timer 1
SETWDON	Set watchdog timer
SETWDOFF	Watchdog off
Variable commands:	
SETH=K	Write constant to variable
SETH=H	Write variable to variable
SET[H]=H	Write variable to indirectly addressed variable
SETH=[H]	Write indirectly addressed variable to variable
SETH=T ∅	Write timer ∅ to variable
SETH=T1	Write timer1 to variable
SETH=AP	Write actual position to variable
SETAP=H	Write variable to actual position
SETH=A12	Write analog values to 2 variables
ADDHK	Add constant to variable
ADDHH	Add variable to variable
SUBHK	Subtract constant from variable
SUBHH	Subttact variable from variable
MULHK	Multiply variable by constant
MULHH	Multiply variable by variable
DIVHK	Divide variable by constant
DIVHH	Divide variable by variable
Teach commands:	
TEACHS	Teach line number
TEACHT	Teach table position
Touch probe commands:	
SETTP	Activate touch probe
GOTPH	Position to touch probe
JMPNTP	Jump if no touch probe
Miscellaneous commands:	
NOP	No operation
CALL	Subroutine call (unconditional)
RET	End of subroutine
END	End of program
LOOPB	Beginning of loop
LOOPE	End of loop
SAVE	Save program to non-volatile memory
BRAKE	Brake function ON/OFF
STOP	Stop drive axis (rapid stop ramp)
COMMENT	Comment line

7.6 Detailed description

7.6.1 Positioning commands

- Positioning with/without wait:

The absolute position and the travel distance can be entered as either a fixed value or a variable. Basically, “wait” and “non-wait” commands can be given. With non-wait commands, program execution continues after the start of travel. This allows parallel processing during travel. A typical application for “non-wait positioning” is a stepped speed profile, without stopping.

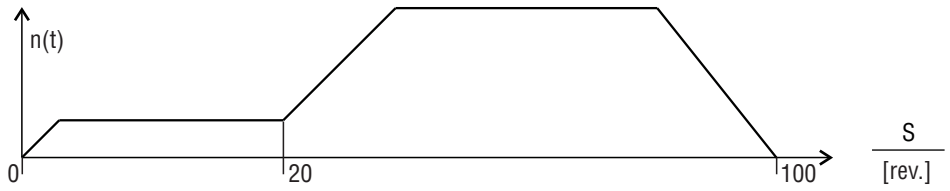


Fig. 20: Positioning without wait 00019ADE

The sample program for this is as follows:

```
GOØ             #Ø                          carry out reference travel
GOWA            #Ø rev.                     move to position Ø rev., wait!
SETNMAX         #1ØØ rpm        #1ØØ rpm     travel speed = 1ØØ rpm
GOA             #1ØØ rev.                   travel 1ØØ rev., non-wait
M00:← JMPAP<K #2Ø rev.           MØØ        wait until actual position 2Ø rev.
SETNMAX         #3ØØØ rpm       #3ØØØ rpm   travel speed = 3ØØØ rpm
WAITPOS                                     wait until axis is at 1ØØ rev.
END
```

Travel takes place between the positions X = 0 rev. and X = 100 rev. At first the speed is 100 rpm. When the position 20 rev. is passed, the axis will travel the remainder of the travel distance at 3000 rpm. The “return travel” is also at 3000 rpm.
The distance units of the position arguments result from the machine parameters “Travel distance factor numerator” and “Travel distance factor denominator” (→ Sec. 4). The symbolic description is determined by the “Travel distance indication” machine parameter.

- Relative positioning with/without wait:

Positioning commands for relative positioning are also executed with or without wait. Travel distances can be entered in user distance units (→ Sec. 4).

Example:

```
GOØ             #Ø       reference axis
GOWA            #Ø rev.  move to X = Ø rev.
LOOPB           #1Ø      begin loop (1Ø times)
GOWR            #5 rev.  move relatively by -5 rev., wait
LOOPE                    end of loop
END                      end of main program
```

The drive travels first to position X = 0. Then the drive travels 5 revolutions each time when the loop is repeated (total of 10). A wait command is used in the example. This means that the drive stops after each 5 revolutions (until it is in the position window) and then starts again. The program runs continuously.

- Positioning by table:

256 variables are available for positioning by table, of which 32 variables can be stored in the non-volatile memory. These are saved with the positioning program, permanently stored in the unit and can be modified from the MC_SHELL user interface.

The commands for positioning by table are formatted as follows:

GOWTA	#terminal mask	#table offset	position absolute to table position, wait
GOTA	#terminal mask	#table offset	position absolute to table position, non-wait
GOWTR	#terminal mask	#table offset	position relative to table position, wait
GOTR	#terminal mask	#table offset	position relative to table position, non-wait

#terminal mask

The table positions are selected with binary coded inputs. The used inputs, marked with an "I" must not be adjacent. Do not use any terminals which are already assigned with other functions (e.g. terminal 2.41 = controller inhibit). The "table position pointer" is calculated from the binary numerical value of the applied terminals plus the table offset specified in the command.

#table offset

The table offset allows an offset of the variables being used for the table positions (e.g. with an entry of 15; variables 16 and higher are used as table position.

Example:

GOTA	**#ØØØØØ**	**ØØØ111**	**ØØØØ**	**#Ø**
	reserved	I terminals on the FEA 31../FIO 31.. options (X8.48 ... 51) (X19:52 ... 54)	I terminals in basic unit (X2.41, X3.42 ... X3.47)	Table offset

Significance	high										low
Terminal	54	53	52	51	50	49	48	47	43	42	41
Option	FIO 31..			FEA 31../FIO 31..				Basic unit			

Terminal	54	53	52	51	50	49	48	47	43	42	41
Terminal mask	0	0	0	0	1	1	1	0	0	1	0
Significance					↓ 2^3	↓ 2^2	↓ 2^1			↓ 2^0	
Terminal level	0	0	0	0	1	0	1	1	0	1	1
Weighting					↓ $1x2^3$	↓ $0x2^2$	↓ $1x2^1$			↓ $1x2^0$	
Table pointer	Table offset + $1x2^3$ + $0x2^2$ + $1x2^1$ + $1x2^0$ = table offset + 11										

4 terminals were specified in the terminal mask which means that 16 table positions can be selected. The table offset allows these 16 positions to be placed in the variable range (0 ... 255). In most cases however, table positions are permanently stored to the varables range 0 ... 32.

- Positioning with variables

The following commands are available for positioning to variable values:

GOWAH H[XXX] travel absolute to position in variable XXX, wait
GOAH H[XXX] travel absolute to position in variable XXX, non-wait
GOWRH H[XXX] travel relative to position in variable XXX, wait
GORH H[XXX] travel relative to position in variable XXX, non-wait

Variables 0 to 255 are available. Except for variables 0 to 31, the variables are not stored permanently (data is lost when the supply is switched off). Variables 0 to 31 can be stored permanently, using the "SAVE" command or "SAVE to EEPROM" menu item. The value range of the variables is -2^{31} to $+2^{31}$. The variables can be read and written to with the user interface. The variables can be edited using the variable commands.

Note: Unlike the other positioning commands, the unit of the variable values for positioning is always 4096 increments/motor revolution (observe sign)!

- Positioning to PC setpoint

The **GOPA** command allows travel within an IPOS program to a position setpoint that has been sent via the serial interface from the user interface (or from a host controller). The command is non-wait but can be extended with the WAITPOS command. Travel takes place to the target position using the set values for the positioning ramp and the travel speed. If more than one setpoint is to be given from a host controller (or the user interface) via the serial link, this can be done by writing the values into a variable. The setpoint for the **GOPA** command is non-retentive and is lost when power goes down.

Example:
```
GOØ #Ø
GOPA
END
```

The program first references the drive axis. Travel then takes place to the position setpoint specified from the user interface or the host controller. The program runs continuously. A new position setpoint, if different, will be employed in the next program run.

The **GOØ** command initiates a reference travel in accordance with the programmed reference travel type and reference travel speed. During a reference travel, the system quits the IPOS operating mode. The GOØ command has an argument, and, depending on this argument, the command has the following effect:

GOØ #0	with GOØ#Ø the drive is only referenced once, afterward the command will be ignored until the system is powered down or the inverter trips.
GOØ #1	with GOØ#1 a reference travel is performed each time when executing the command.

The **NOP** command has no effect in the program but the processing time for this command is also 5 ms.

7.6.2 Jump commands

General: Every command can be extended with a label using the MC_SHELL user interface. This label is then used as the jump destination. The **basic programming rules** described in Sec. 7.1.1 apply to the use of jump commands. These must be strictly observed for trouble-free operation.

- Unconditional jump:

The command **JMP M[XX]** implements an unconditional jump to label XX.

- Jump if drive stopped

The command **JMPST M[XX]** will jump to label M[XX] if the drive is stopped. If the drive is not stopped, the command that follows will be executed.

- Jump if drive not in position

The command **JMPNPOS M [XX**] will jump to label M [XX] if the drive is not in position. If the drive is in position, the command that follows will be executed.

- Jump if input is high or low

The commands **JMPI1** and **JMPIØ** are available. The command format is as follows:
JMP1 1/0 #terminal mask M[XX]
In the terminal mask, all physical terminals that are to be used for jump conditions must be set to 1. Unused terminals must be set to 0. When the **JMPI1** command is executed, a jump to Label[XX] takes place, if **all** the physical terminals set with 1 in the terminal mask have a level of 1. If this condition is not met, the next command will be executed. The same applies to the **JMPIØ** command, except that all the terminals set with 1 in the terminal mask must have a level of 0 in order for the jump condition to be true.

Terminal mask format = (→ Sec. 7.6.1 Positioning by table)
The lowest numbered terminals are on the right.

```
Example:   GOØ      #Ø
           JMPI1    #ØØØØ ØØØØ ØØ11 ØØØØ,  MØ1
           WAIT     #1ØØØ    ms
    MØ1:   GOWR     #1Ø      rev.
           END
```

The drive travels relatively by 10 revolutions each program cycle (after referencing). If **both** the first two input terminals of the FEA 31../FIO 31.. are at level 1 (high), the command WAIT 1000 (1 s) will be ignored. The program runs continuously.

- Jump dependent on timer

Two timers are available, each having a setting range of 0 ... 30s. After the timer has been set (see Set commands), it times down (to 0) during program execution.
The commands **JMPTØ=Ø** and **JMPT1=Ø** execute a jump to the given label if timer Ø or timer 1 has elapsed. If the timer has not yet timed out, the next command will be executed.
Example:

```
           GOØ              #Ø
           GOWA             #Ø            rev.
           SETTØ            #1ØØØØ ms
           GOWA             #5ØØ          rev.
           JMPTØ=Ø          MØ1
           SETO1            #ØØØØ ØØØ1
           JMP              MØ2
    MØ1:   SETOØ            #ØØØØ ØØØ1
    MØ2:   RET
           END
```

The drive first positions to travels 0 rev. (after reference travel). Timer 0 is set to 10 s. If the drive has approached the position 500 rev. and the 10s have not expired, the first output terminal goes high (1 signal). If the time has expired after positioning, the program jumps to MØ1 and the first output terminal is set to Ø.

- **Jump dependent on value of variable:**

The following jump commands dependent on variable values are available:

JMPH>K jump to label if variable greater than constant
JMPH<K jump to label if variable less than constant
JMPH=K jump to label if variable equal to constant
JMPH>H jump to label if variable greater than variable
JMPH<H jump to label if variable less than variable
JMPH=H jump to label if variable equal to variable

The value range of both constants and variables is -2^{31} to $+2^{31}$. Negative signs are considered in evaluating the conditions.

Example:

```
          GOØ     #Ø
          SETOØ   #11111111            all outputs = Ø
          GOWA    #Ø rev.              first travel to X = Ø
          SETTØ   #10000 ms            set timer to 1Ø s
   MØØ:   SETH=TØ HØØ1, TØ             copy current timer content to variable
          JMPH>K  HØØ1, #8000,   #MØØ  jump back if 2 s not expired
          SETO1   #0000 0001            switch on output 1 after 2 s
          JMPH>K  HØØ1, #6000,   #MØØ  jump back if 4 s not expired
          SETO1   #0000 0010            switch on output 2 after 4 s
          JMPH>K  HØØ1, #4000,   #MØØ  jump back if 6 s not expired
          SETO1   #0000 0100            switch on output 3 after 6 s
          JMPH>K  HØØ1, #2000,   #MØØ  jump back if 8 s not expired
          SETO1   #0000 1000            switch on output 4 after 8 s
          END
```

The program first deactivates all output terminals and travels to X = 0. The timer is set to 10s. Every 2s an output terminal goes high (1 level). When the last output has been set, the program begins again.

- Jump dependent on the actual position

The commands **JMPAP>K** and **JMPAP<K** execute a program jump to a given label if the current actual position is greater or less than the given absolute value.

Command syntax:

JMPAP>K #[travel distance entry] M[XX]
JMPAP<K #[travel distance entry] M[XX]

The travel distance must be entered in user distance units (constant).

The commands JMPAP>H and **JMPAP<H** execute a jump to the given label if the current actual position is greater or less than the value in the specified variable.

Command syntax:

JMPAP<H H[XXX] M[YY]
JMPAP<H H[XXX] M[YY]

It should be noted that position-related variable values are **always** in the unit 4096 increments/motor rev. The setting of the machine parameters “Travel distance factor numerator” and “Travel distance factor denominator” have **no** influence on the unit of the variable.

7.6.3 Wait commands

The Wait commands prevent further execution of the program until the specified condition is met. The program then continues with the next command. The following conditions are available:

WAIT #[time] ms direct time entry, units ms, range 0 ...30000ms
WAITI1 #[terminal mask] wait until **all** physical terminals marked with 1 are high (at level 1).
WAITIØ #[terminal mask] wait until **all** physical terminals marked with 1 are low (at level 0).
WAITPOS wait until the drive approaches the position window.

7.6.4 Set commands

The commands **SETOØ** and **SETO1** allow a maximum of 8 output terminals to be assigned. However, only 3 outputs on the FEA 31../7 outputs on the FIO 31.. are available for programming. All terminals marked with 1 in the argument of the two commands will be set active or deactivated. The terminals marked with 0 remain unaffected.

Command syntax: SETOØ [X8 X7 X6 X5 X4 X3 X2 X1]
SETO1 [X8 X7 X6 X5 X4 X3 X2 X1]

Argument	X1	X2	X3	X4	X5	X6	X7	X8
Pos. output	no. 1	no. 2	no. 3	no. 4	no. 5	no. 6	no. 7	no. 8

To enable the **output terminals** to be physically activated from IPOS, the desired output must be set to the function pos. output no. 1 ... 8 in parameters **61 .. 617**. Do not use any terminals which are already assigned with other functions (e.g. terminal X3.61 = brake function).

Example:

```
GOØ          #Ø
WAIT         #1ØØØ  ms
SETO1        #ØØØØ 1Ø1Ø
WAIT         #1ØØØ  ms
SETOØ        #ØØØØ 1Ø1Ø
END
```

In the example, the logical terminals Output IPOS 2 and Output IPOS 4 are set or deactivated respectively at the same time after 1s. The logical terminals Output IPOS 1, 3, 5, 6, 7, 8 remain unaffected by this program. If, for example, Output IPOS 2 on connector X3.62 of the MOVITRAC®31.. (→ MOVITRAC® 31.. Operating Instructions) is now to be activated, Output IPOS 2 must be set in menu item “Terminal assignment”, parameter 611.

- Setting of travel-specific parameters

Within the positioning program, the travel speed and positioning ramp (Machine parameters →Sec. 4) can be changed, also while the drive is traveling.

Note: When adjusting travel speed or positioning ramp, the system conditions must be taken into account, in particular motor rated speed and shortest possible positioning ramp. The unit itself **does not** limit the setting ranges.

The command syntax is:

SETNMAX #[travel speed CW], #[travel speed CCW]
Entry in rpm, setting range: CW = 1 rpm ... 5000 rpm
CCW = 1 rpm ... 5000 rpm

SETNMAX=H[XXX]
Entry in rpm, setting range: H[XXX] for CW = 1 rpm ... 5000 rpm
following variable for CCW = 1 rpm ... 5000 rpm

SETAMAX #[positioning ramp]
Entry in ms, setting range: 0 ms, 20 ms ... 10000 ms

SETAMAX=H[XXX]
Entry in ms, setting range: 0 ms, 20 ms ... 10000 ms

After executing the set commands, the changed values for the **travel speed CW/CCW and positioning ramp** parameters are displayed in the user interface. The machine parameter values changed by the set commands are retained until the unit is switched off or until new values are set through the user interface or if a SAVE command is used in the program.

- Setting the timers

The timers can each be set with an initial value using the commands **SETTØ** and **SETT1**. The timers count down until they reach the value 0. The setting range is 0 ms ... 30000 ms.
Command syntax:

SETTØ	#[time value]	*variable not possible*
SETT1	#[time value]	*variable not possible*

- The command SETWDON activales a time monitoring function for the positioning program and sets/resets the monitoring time to the given value.

Command syntax:
SETWDON #[time value]
SETWDOFF
If the watchdog time elapses before the monitoring function is turned off (SETWDOFF) or a new time is set, the system trips with F56 (fault watchdog) and the drive is brought to a stop.

The **SETWDOFF** command allows the watchdog processing to be stopped. The time monitoring does not begin again until the **SETWDON** command is executed.

7.6.5 Variable commands

It is not possible to access the variables from the FBG 31.. control keypad.

- Writing to variables

256 variables (H000 ... H255) can be written with: **constant values**, other **variables**, **timers** and the **actual position**.

The value range of all variables is -2^{31} to $+2^{31}$. Except variables 0 ... 31 the variables are only stored in volatile memory, their values are lost if the unit is powered down. Variables 0 to 31 can be saved using either the SAVE command in the automatic program or by using menu item "SAVE to EEPROM" after a program download (F2).

Command syntax:

SETH=K	H[XXX],	#[constant]
SETH=H	H[XXX],	H[YYY]
SET[H]=H	[HXXX],	H[YYY]
SETH=[H]	H[XXX],	[HYYY]
SETH=TØ	H[XXX],	TØ
SETH=T1	H[XXX],	T1
SETH=AP	H[XXX],	Xactual
SETAP=H	Xactual,	H[XXX]
SETH=A1 2	H[XXX],	analog input 1 / analog input 2

The commands **SET[H]=H** and **SETH=[H]** can be used to address all variables indirectly by setting or increasing up the variable in brackets accordingly. The value of the variable not in brackets is written to the indirectly addressed variable.

Example:

GO Ø	#0	
SETH=K	H000, #0	
LOOPB	#16	(→ section 7.6.8)
GOWR	#10000 incr.	
SETH=AP	#65, AP	
SET[H]=H	[H000], H65	
WAIT I1	# 0000 0000 0000 0100	
LOOPE		(→ section 7.6.8)
END		

The drive first performs a reference travel and variable H000 is set back to zero. Within the program loop (LOOPB, LOOPE) the drive then travels 10000 incr. per cycle. The new actual position is written to an indirectly addressed variable. The variable address is increased by one every cycle. In the program the actual position is written to variables H001 .. H016.

The **SETH=A1/2** command allows the values of the two analog inputs to be read to variable values. Analog input 1 (X7:32) → voltage range 0...+10 V (0..+1000), analog input 2 (x2:34) → voltage range -10 V ... + 10 V (-1000 to +1000). The argument to be entered represents the number of the variable for analog input 1 of the MOVITRAC®31.. . The value of analog input 2 is written to the next successive variable. The display values of the analog inputs are filtered with a time constant approx. 200 ms.

- Operations with variables

The four basic types of calculation can be implemented with these variables. **Constants** or other **variables** can be used as the second operand.

The result of the operation is assigned to the variable (left operand).
Command syntax:

ADDHK	H[XXX],	#[constant]
SUBHK	H[XXX],	#[constant]
MULHK	H[XXX],	#[constant]
DIVHK	H[XXX],	#[constant]
ADDHH	H[XXX],	H[YYY]
SUBHH	H[XXX],	H[YYY]
MULHH	H[XXX],	H[YYY]
DIVHH	H[XXX],	H[YYY]

7.6.6 Teach commands

The **TEACHS** (teach line) and **TEACHT** (teach table) commands can be used to "learn" position values as new target positions without a PC connected. The command syntax is:
TEACHS M[XX]
TEACHT #terminal mask #table offset (variable offset)
When using a Teach command a TEACH TERMINAL has to be specified in the machine parameters. (Do not use any terminals which are already assigned with other functions).
The teach terminal can be set from 1 to 10. This means:

Setting value of teach terminal (machine parameter)	Resulting terminal activated as teach terminal	
1	X3.42	Basic unit
2	X3.43	Basic unit
3	X3.47	Basic unit
4	X8.48	FEA 31../FIO 31.. option pcb
5	X8.49	FEA 31../FIO 31.. option pcb
6	X8.50	FEA 31../FIO 31.. option pcb
7	X8.51	FEA 31../FIO 31.. option pcb
8	X19.52	FIO 31.. option pcb
9	X19.53	FIO 31.. option pcb
10	X19.54	FIO 31.. option pcb

If the **level of the teach terminal** is Ø at the **time of command execution (TEACHS or TEACHT)**, the teach command has **no** effect. If the **level is 1** when the command is executed, the **teach mode becomes active**.

In the teach mode the analog setpoint 2 (X2.34 / X2.35) is used as speed setpoint. The drive can now be moved to the desired position. To stop more accurately and hold the drive drift-fee in the desired position, the following characteristic of the speed setpoint is active during the teach process.

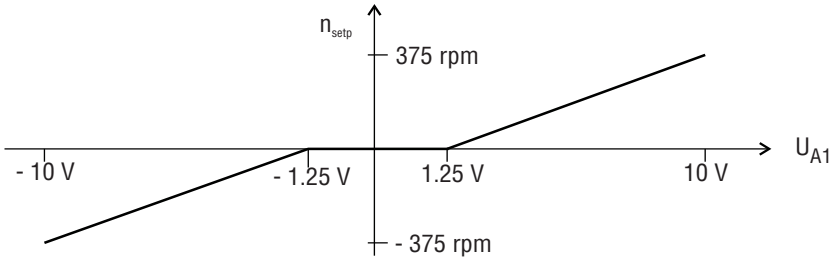


Fig. 21: Speed setpoint characteristic MD0077AEN

To achieve precise manual positioning no speed setpoint is generated within the voltage range of -1.25 V ... 0 ... +1.25 V. The maximum manual teach speed is ± 375 rpm.
To "learn" the present actual position, the TEACH terminal has to be set low (level 0). The **actual position** is written into the argument of the corresponding positioning command **(TEACHS)** and to the variable selected in the terminal mask **(TEACHT)**. A jogging function as part of the automatic program may also be used to travel to the desired position. (→ Sec. 8.2).

With the **TEACHS** command the present actual position, "learned" in the teach mode is written to the argument of the positioning command. Both, the TEACHS command and the positioning command have to be extended with the same jump label, as shown in the example below:

Example:

```
          GOØ             #Ø
M03 GOWA  #Ø              rev.
          WAIT            #1ØØØ  ms
          TEACHS          MØ3
          GOWR            #25    rev.
          SAVE            (→ Sec. 7.6.8)
          END
```

In the sample program, the target position in the argument of the GOWA command will be the position "learned" with the TEACHS command. To do this, the level on the teach terminal must be 1, so that the analog setpoint can be used to travel to a certain position. This position is "learned" into the argument of the GOWA command when the level on the teach terminal changes from 1 → 0. To display the "learned" argument in the position window, the window must be closed and then reopened again. The positioning program and the newly "learned" target position are saved in the background (parallel to the running program) with the SAVE command.

The **TEACHT** command, allows to write position values to a table (variables H000 ... H031) in the same procedure. The terminal mask shown on page 32 describes which input terminal is used as a "pointer" within the variable table. The table offset (variable offset) is added to the current "variable pointer".

Example:

```
GOØ           #Ø
TEACHT         #ØØØØ ØØØØ 1111 ØØØØ #Ø
SAVE          X8.51 ... X8.48        variable offset
               2^3 ... 2^0
JMPIØ          ØØØØ ØØ1Ø ØØØØ       ØØØØ, MØØ
GOTA          #ØØØØ ØØØØ 1111  ØØØØ #Ø
END
```

This program allows 16 table positions (variables 0 to 15) to be read in with the TEACHT command. The 4 input terminals X8.48 to X8.51 serve as a pointer for the table positioning command GOTA and as a pointer for the TEACHT command. If the teach terminal selected in the machine parameters remains at a level of 0, the drive will travel to the table position selected with the terminals if the GOTA command is used. The program runs continuously. The positioning program and the newly "learned" table positions are saved in the background (parallel to the running program) with the SAVE command.

7.6.7 Touch probe commands

SETTP: When using the Touch Probe function input X3.42 is automatically fixed assigned as the TP input and has to be programmed with "No function".
With the TP-logic active (by SETTP) the next 1 level on X3.42 causes the present actual position to be stored as the "Touch probe position". The TP signal (level 1) must be present for a minimum of 1 ms to be recognized reliably. Only the first recognized 1 level (on X3.42) after execution of the **SETTP** command will cause the actual position ("touch probe position") to be saved. The IPOS program is continued after the **SETTP** command, whether the touch probe (1 level on X3.42) was received or not. For further processing, the touch probe position is stored in the IPOS variable 255.

GOTPH: The **GOTPH** command "waits" until the touch probe has been received. The command has a variable number as its argument. After the touch probe has been received, the new target position is calculated as "touch probe position + content of the variable". This command is non-wait but can be extended with WAITPOS.

JMPNTP: The command "Jump if no touch probe signal is recognized" can be used, to loop the program until a TP-signal is received.

Sample program:

```
MOO:       GO0     #0             reference axis
           SETO0   #0000 0001     reset signal output
           GOWA    #0             go to x=0 wait
           GOA     #100 rpm       go to 100 rev. non-wait
           SETT0   #2000ms        start timer with 2000ms
           SETTP                  activate touch probe
M01:       JMPT0=0 #M02           2s expired without touch probe  ?
           JMPNTP  #M01           touch probe recognized?
           GOTPH   H005           position absolute to TP position + value
                                  specified in variable H005
           WAITPOS                wait until target position is reached
           JMP     #M00           repeat cycle
M02:       SETO1   #0000 0001     signal that no touch probe was recognized
           JMP     #M02
           END
```

7.6.8 Miscellaneous commands

The **CALL** command executes an unconditional jump to a subroutine, specified with a jump label.
Command syntax:

CALL M[XX]

The subroutine ends with a return command RET and execution of the main program continues with the next command following the CALL command. Nested subroutine calls are possible. When the CALL command is used, the basic programming rules in Sec. 7.1.1 must be observed!

The **RET** command identifies the end of a subroutine. The complete program is ended with the **END** command (→ Sec. 7.2).

The commands **LOOPB** and **LOOPE** allow repeated processing of a command block. The command syntax is:

```
LOOPB       #[number of loop cycles]
Command
Command
...
...
LOOPE
```

All commands within the loop block are executed for the number of loops specified in the argument of the command LOOPB. Then the next command is executed. The loop command can be nested. The loop block cannot be exited by a jump. The basic programming rules in Sec. 7.1.1 must be observed.

With the **SAVE** command the positioning program and the permanently storable variables 0 to 31 are permanently stored in the inverter (EEPROM). This process takes about 15s and is executed parallel to the program. This command is useful when the program has been modified, e.g. using the TEACH mode.

Note:
Only **changes** in the program and the variables 0 to 31 are saved in the non-volatile memory by the **SAVE** command. This memory has a limited number of storage cycles (approx. 10^5 ... 10^6). If variables are changed continuously (e.g. for entering setpoints from a serial interface or by the SET H = A12 command or as calculation variables), then the **SAVE** command should not be used cyclically in the program.

The brake function can be activated permanently via the machine parameter “Brake function = ON” or switched on and off in the positioning program. The argument of the brake command is “1” to set the brake and “0” to release the brake.

The **STOP** command can be used to stop the axis. If the **STOP** command is executed, the drive is brought to a stop (with the rapid stop ramp). As soon as the drive is at rest, position control is initiated and program execution continues.

A **COMMENT** line can be entered in the program by selecting the command COMMENT in the list of commands. COMMENTS are saved to disk or hard drive only. After an upload from the inverter all comment lines are lost.

8 Application Notes

- **Starting IPOS via the terminals**

To start IPOS only via the input terminals, the last PC operating mode setting in the positioning window must be START while the controller is inhibited and enabled (both terminals level 1). Otherwise this operating mode must be activated with the F9 function key. The parameter EEPROM memorizing (P801) has to be set "ON".

- **Feedback devices**

Supported encoder systems are incremental encoders with 128/256/512/1024/2048 pulses per revolution. IPOS always operates internally with 4096 increments/rev., regardless of the number of pulses of the incremental encoder.

- **Machine parameters**

Determine the machine parameters before you write the program, as they influence e.g. the travel distances specified in the program.

- **Deactivate monitoring functions**

The machine parameters timeout period, lag error and software limit switches (only when both limit switches = 0) are deactivated by the value 0.
Observe the appropriate Safety Notes.

- **Variables as distance values**

The travel distance entered as a variable is always in unit increments (4096 incr./rev.) where as a travel command with a fixed value is entered in user distance units.

- **Travel distance entries with decimal points**

If the fraction travel distance factor numerator/denominator is a non-integer value, the reference offset, software limit switch values and the travel positions in the program are also nin-integer values which are processed by the program but rounded (see example for a hoist application). To be more accurate the numerator and denominator can be multiplied by a factor (e.g. 1000) to eliminate the decimal portion.

- **Motor rated speed**

The travel speed can be set equal to the motor rated speed for IPOS mode.

- **In-position signal**

If an output terminal is programmed with the function "In position", this output turns high as soon as the drive approaches the position window (specified as machine parameter) when executing a travel command. As soon as the drive exits the position window, the output "In position" turns low (level 0).

- **Reference travel type 3/4**

If the drive is located on the limit switch (which serves to reference the drive) when turning on the system and the inverter is inhibited (X2.41 = 1) the drive clears the limit switch using the ref. speed V_{ref2} and then restarts the program.

8.1 Hoist application

8.1.1 Schematic structure

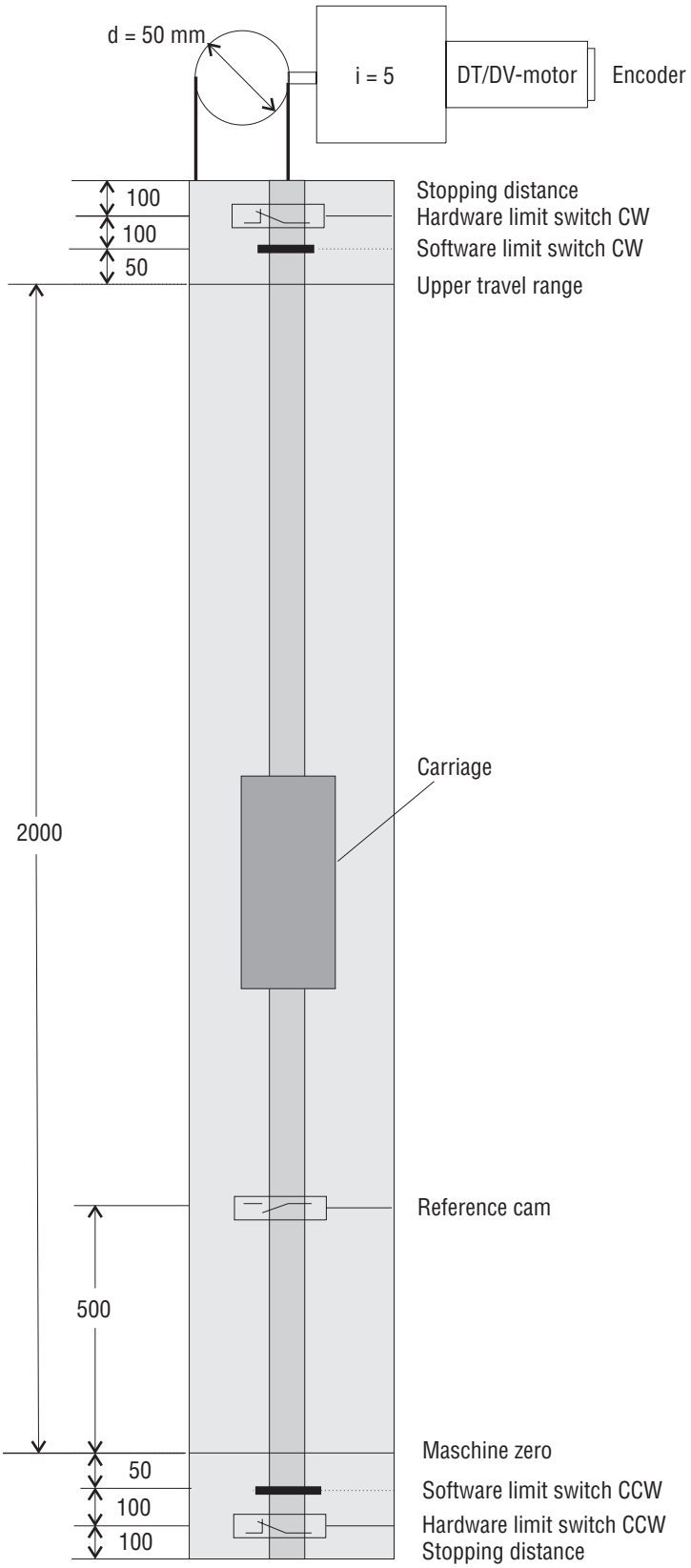


Fig. 22: Schematic structure of a hoist application

8.1.2 Terminal assignment

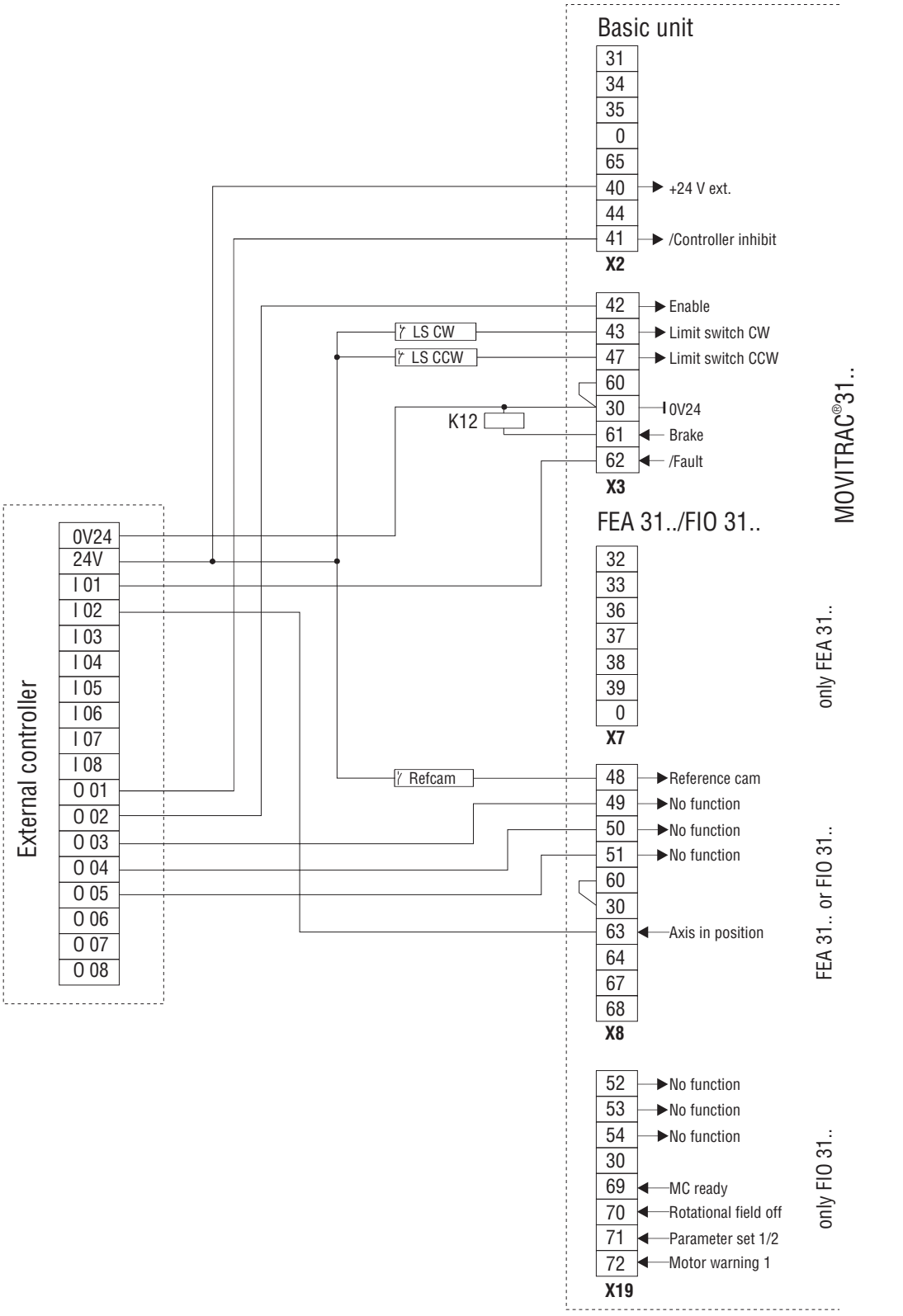


Fig. 23: Terminal assignment in a hoist application

8.1.3 Setting MC_SHELL parameters for IPOS applications

770	**Control functions**	
770	Operating mode	POSITIONING
324/202	**Limits**	
324	Pole pair number	2
202	f_{max}	55Hz
	This gives a max. speed of 1650 rpm.	
60.	**Binary inputs - basic unit**	
600	terminal X3.42	Enable
601	terminal X3.43	/LS CW
602	terminal X3.47	/LS CCW
60.	**Binary inputs - FEA 31..**	
603	terminal X8.48	REF. CAM
604	terminal X8.49	NO FUNCTION
605	terminal X8.50	NO FUNCTION
606	terminal X8.51	NO FUNCTION
611	**Binary outputs - basic unit**	
611	terminal X3.62	/FAULT
612	**Binary outputs - FEA 31..**	
612	terminal X8.63	IN POSITION

IPOS machine parameters

Reference offset		499.99453711
Reference speed 1	rpm	500
Reference speed 2	rpm	100
Reference travel type		1
Gain position controller		2
Positioning ramp	s	1
Travel speed CW	rpm	1500
Travel speed CCW	rpm	1500
SW limit switch CW	mm	2049.993709
SW limit switch CCW	mm	-50.000220703
PC pos. setpoint	mm	0
Position window	inc	50
Override		OFF
Teach terminal		0
Lag error window	incr	5000
Travel distance indication		mm
Travel distance factor numerator		2048000
Travel distance factor denominator		15708
IPOS bus mode		0
Feedforward	[%]	50

8.1.4 Calculating IPOS machine parameters

Reference offset: See schematic
SW limit switch: See schematic

Travel distance indication: Travel-specific data shall be in mm.

Travel distance factor numerator: The travel distance unit shall be mm!
Number of increments per drive wheel revolution
Increments/motor rev. • gear ratio
4096 increments • 5 = 20480
20480 • 100 (extension factor to higher terms*) = 2048000

Travel distance factor denominator: Drive wheel perimeter in mm
d • π
50 mm • = 157.0796327
157.08 • 100 (reduction factor*) = 15708

* An extension factor may be used to eliminate the decimal portion.

Feedforward: Only 50%, in this application no absolutely linear ramps are to be travelled (soft run-in to target position), i.e. easy on the mechanical components.

Travel speed: Motor rated speed

Position window: At 50 increments the 'Drive in position' message shall be issued
(drive will always stabilize to ± 1 increment).

8.1.5 Hoist program

```
     GOØ      (#Ø)                                carry out reference travel
MØ2: JMP IØ    (ØØØØ), ØØØØ, ØØ1Ø, ØØØØ, MØØ     when terminal X13.3 set, go
     GOWA     (Ø mm)                              to position 0 mm
MØØ: JMP IØ    (ØØØØ), ØØØØ, Ø1ØØ, ØØØØ, MØ1    when terminal X13.4 set, go
     GOWA      (999.98907421 mm)                  to position 1000 mm
MØ1: JMP IØ    (ØØØØ), ØØØØ, 1ØØØ, ØØØØ, MØ2     when terminal X13.5 set go
     GOWA      (1999.9858183 mm)                  to position 2000 m
     RET
     END
```

8.2 Sample program for jogging application

Jogging mode for continuous travel (no counter overflow when 2^{31} incr. is reached)

Example:

```
****** Jogging / Continuous travel *******
    SETNMAX         R#300    , L#300       rpm
M02:JMPI1           #0000 0000 0001 0000,  M00
    STOP
M03:JMPI1           #0000 0000 0010 0000,  M01
    STOP
    JMP             M02
****** Jogging CW ***********************
M00:SETH=AP         H1       , AP
    ADDHK           H1       , #40960
    GOAH            H1
    JMP             M02
******Jogging CCW *********************
M01:SETH=AP         H1       , AP
    SUBHK           H1       , #40960
    GOAR            H1
    JMP             M03
    END
```

This jogging program allows a motor to travel infinitely without generating a *travel distance counter overflow* when exceeding 2^{31} increments. To implement this, the **travel distance variable** is determined during travel based on the **present actual positon + an offset** throughout the program cycle. IPOS allows the travel distance variable to be set higher than 2^{31}, changing over to 2^{-31} (→ Fig. 24). If the variable is increased even further, the counter will eventually return to the positive section, thus allowing infinite motor travel. To ensure jerk-free smooth motor travel at max. speed an offset of at least 10 motor revolutions (40960 increments) must be added cyclically to the current position.

Note:
If the *GOR* command is given during the program cycle, this may cause the setpoint position to "run away" from the actual position of the motor and cause the control to trip when a difference of $2^{31}/2$ is reached.

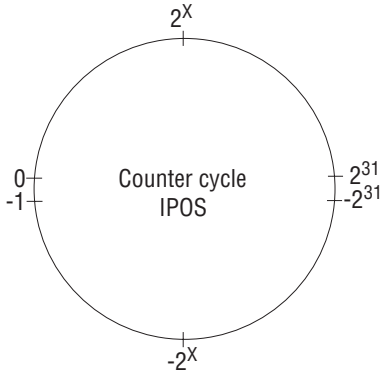


Fig. 24: Counter cycle

00400AEN

9 Index