LinMotâ

User Manual Rel. 1.0

LinMot
Sulzer Electronics AG
Technoparkstrasse 1
CH-8005 Zürich

Tel.: +41 (0) 1 445 2282
Fax: +41 (0) 1 445 2281
Email: office@linmot.com
Homepage: www.linmot.com

Note
The information in this documentation reflects the stage of development at the time of press and is therefore without obligation.
Sulzer Electronics reserves itself the right to make changes at any time and without notice to reflect further technical advance or product improvement. Please refer to the latest edition of our “General business terms”

Version 1.01 / Rev. 3 October 8th 1999

1. Introduction

This manual is organized as follows:

Safety notes

Please take a moment to read and get accustomed with the safety instructions. This information is intended for your personal safety and should help in protecting the electronic units and linear motors from possible damage.

System description

Gives an overview of the properties and features of the single *LinMot*® components and their operation.

Design and installation

Here you will find a description of the functionality, the different operation modes and the technical data of the electronic unit. Further, the interfaces and the installation are described in this chapter. This information helps you optimally embed the *LinMot*® products in your application.

Configuration software

In this chapter you will get accustomed with the functionality and possibilities of the PC software for the configuration of the *LinMot*® electronic unit. Further, you will find examples that will make the first commissioning of the system easy.

Parameters

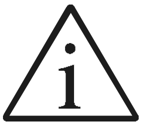
This chapter gives an in-depth description of all the parameters to which you have access during the configuration of the *LinMot*® electronic unit.

Service

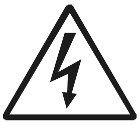
Explains the error messages that are displayed by the LEDs and helps you in the troubleshooting.

1.1 Used symbols

Important notes or tips in this manual are marked with the following symbols:



Tips, notes
Useful information that should make operating the devices easier is given here.



Caution
Non observance of these warnings can represent a danger for health or life and lead to damage or destruction of the devices and other objects.

Strong magnetic fields
Warning about magnetic materials that could cause damage to magnetic data mediums like diskettes, credit cards, etc.

2. Safety notes

The *LinMot*® linear drives are resources meant to be built into an electrical systems or machines. During operation these resources have moving parts and hot surfaces from which a danger of health or material damage could arise.

The commissioning (putting the device into operation as specified in the application) of the linear motors is prohibited, as long as the machine does not comply with the relevant safety prescriptions.

Those responsible for the safety of the plant or machine must guarantee that only qualified personal accustomed to working with electrical drive equipment is allowed to work on the devices in order to prevent and avoid injuries to persons and damage to the equipment.
Qualified personal are those who, as a result of their training, experience and instruction as well as their knowledge of the relevant norms, regulations, accident prevention prescriptions and service conditions have been entitled by the responsible for the safety of the plant to execute the necessary work and thereby recognize and prevent possible danger.

The operation manual as well as any further manuals in the documentation of the product are to be followed consequently when executing the respective operations.

Only tested and potential separated power supplies are allowed to be used for the voltage supply of the *LinMot*® electronic units and other accessories.

The electrical installation is to be done according to the relevant prescriptions. Any further prescription contained in the documentation is to be observed.

The linear drives must be protected from loads in excess of the specifications. Particularly during transportation great care must be taken in order to prevent excessive load or even bending of any part.

Electronic devices are basically not fail-safe. The user is responsible for driving the linear drive in a safe state in case of a failure.

Inside the sliders of the linear motors are strong permanent magnets. These can cause damage or corruption to magnetical data mediums such as diskettes, credit cards, etc.

These safety notes imply no claims regarding completeness. For questions and problems please contact Sulzer Electronics AG.

2.1 Installation

The drive systems described in this operation manual are components and not serviceable or connection-ready devices or machines in terms of the device safety laws, the EMC laws or the CE machinery guidelines.

The components are intended to be embedded in other machines. The ultimate mode of action is only defined once these components have been embedded in the user’s machine or plant. The conformity of the construction with the existing prescriptions and laws lies under the sole responsibility of the user.

The commissioning of the components is only permitted in machines or plants that comply with the CE machinery guidelines and/or with the relevant safety prescriptions of the respective country.

3. System Overview

3.1 Drive system *LinMot®*

LinMot® is a modular built servo drive system that can be operated as 'stand-alone' or in connection with commercial PLCs, PCs or other higher-level control systems. *LinMot®* incorporates the basic idea of the peripheral drive elements: the reference values are given directly by the higher-level control system in form of position set values, while the actual motion profiles and position control are calculated and executed peripheral in the *LinMot®* electronic unit. MS-Windows based configuration and commissioning programs release the engineers and draftsmen from electrotechnical detail questions and allow for quick and function oriented work.

Among the actuators connectable to the *LinMot®* electronic units, the electromagnetic direct linear drives *LinMot®* P take a special position, as they offer completely new possibilities in the machine construction and mechatronic system concept.

The notions *LinMot® Step* and *LinMot® Magnet* refer to complementary drive families consisting of the combination of commercially available stepper motors and magnets as actuators and *LinMot®* electronic units. Both stepper motors and electromagnets can be used in mixed system configurations with the *LinMot®* linear drives.

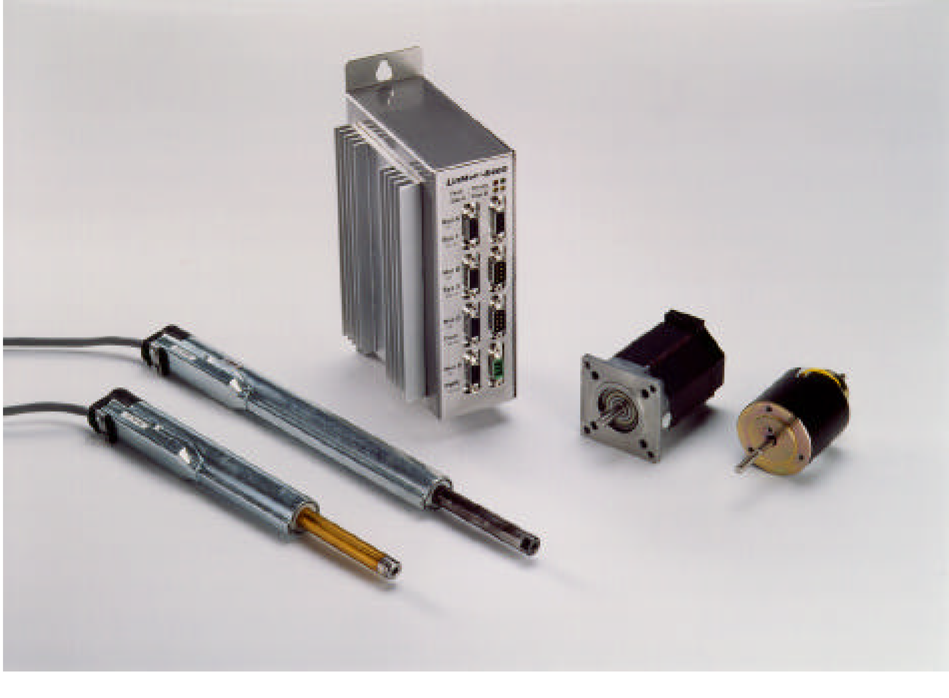


Figure 3-1: Mixed system configuration consisting of two linear motors *LinMot® P*, a *LinMot® E* electronic unit as well as a stepper motor and an electromagnet.

3.1.1 Actuator: Linear drive *LinMot® P*

*LinMot® P*s are electromagnetic direct linear drives with integrated position sensing and bearing. *LinMot® P* linear motors are intrinsically free of mechanical play, gearing or belt wear. The enormous dynamic properties and compact construction of *LinMot® P* make a variety of applications and novel constructions possible.

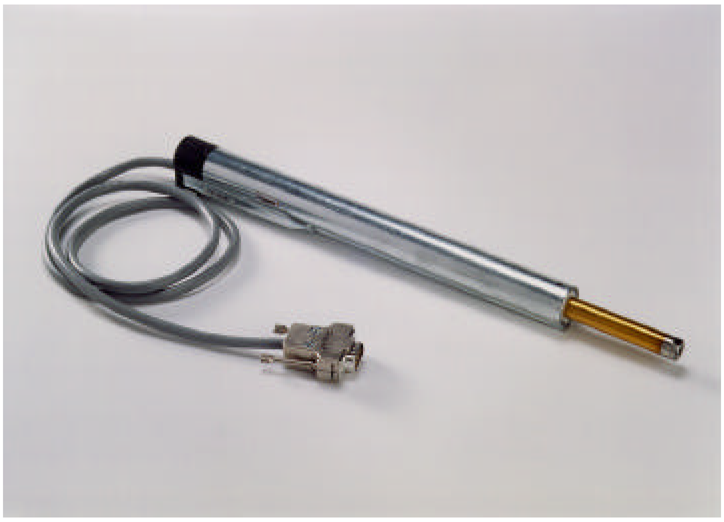


Figure 3-2: Electromagnetic direct linear motor *LinMot® P*

3.1.2 Actuator: stepper motor *LinMot® Step*

LinMot® Step allows the open-loop operation of commercially available two-phase stepper motors with *LinMot® E* electronic units.

3.1.3 Actuator: electromagnet *LinMot® Magnet*

LinMot® Magnet allows the intelligent control of electromagnets as used in short-stroke positioning elements or valves. Compared to the conventional voltage controlled operation of electromagnets, the pull-in and release times of such elements can be greatly reduced with the use of *LinMot® Magnet*, by driving them over-excited during the transient.

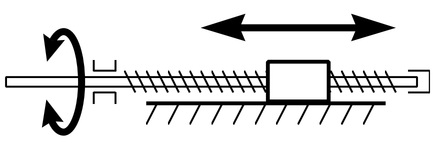
3.2 *LinMot® P*: a new construction element

Most commercially available servo drive systems only dispose of rotating motors, where the speed and in some cases the angular position are controlled. However, many applications don't only need rotational movements, but linear movements too. The conventional solution is to realize mechanical constructions based on levers, cam disks, spindles, belts etc. In which the rotational movement of the servo motor is translated in a linear movement. This way the servo drive serves basically as a 'controlled energy source'.
In this sense the *LinMot® P* linear drives are more than just 'controlled energy sources': *LinMot® P* executes dynamic controlled and position controlled linear movements directly, with no need for additional mechanical gearing elements. *LinMot® P* fulfills the mechatronic basic idea according to which the movement functions are executed directly, with minimal mechanical elements and right in the

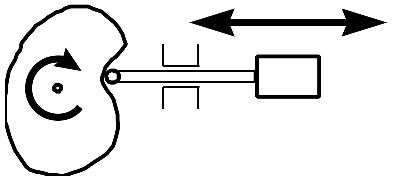
place where they are needed. The motion control, i.e. the actual machine functionality as well as the motion parameters are defined exclusively in the software and can be adjusted at any time with no need for intervention in the mechanic.

3.3 Substitution possibilities of *LinMot*® *P*

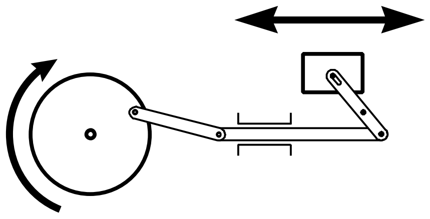
Depending on the application, *LinMot*® *P* can substitute the following conventional solutions and extend them with additional functionality due to its complete electronic control:



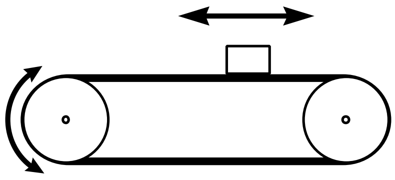
Spindle drives



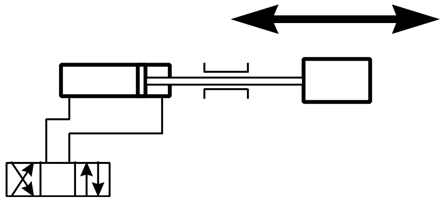
Cam disks



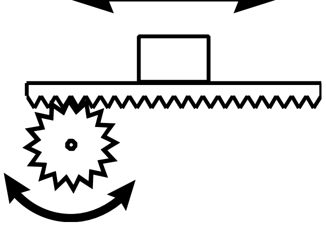
Mechanical levers



Belt drives



Pneumatic cylinders



Rack-and-pinion drives

3.4 Technical properties of *LinMot®* *P*

The linear drives *LinMot®* *P* are characterized by the following properties:

- Connection to commercially available PLC and PC control systems
- Position controlled and curve-driven operation
- Dynamic properties of the drive adjustable (v_{max}, a_{max})
- Extremely high dynamic specification (e.g.: continuous cyclical operation >15 Hz)
- Compact, connection-ready construction with integrated sensing and bearing
- Suitable for operation in rough industrial environments

3.5 Application fields of *LinMot®* *P*

The linear drives *LinMot®* *P* can be used both as stand-alone systems or in connection with highly complex machines. The following list of application examples is therefore not complete and gives only an insight in the variety of the possible applications:

- **Textile machines**
  Weaving machines, knitting machines, carpet machines, winding machines
- **Packaging machines**
  Food: pastries, coffee, tee, sweets
  Consumer goods: office material, toys
  Cosmetics: soap, tubes
  Pharmaceutical: tablets, pills
- **Printing machines**
  Offset and tampon printing machines
- **Assembly systems**
- **Winding machines**
- **Handling machines**
- **Robotics**
- **Laboratory automation**
- **Folding machines**
- **Labeling machines**
- **Newspaper and paper machines**
- **Sorting machinery**
  Parcel and letter sorting systems, material sorting machines
- ….

3.6 Driving and operating modes of *LinMot*®

3.6.1 Driving concept

The basic idea of *LinMot*® *P* consists in having the control tasks executed as peripheral as possible in order to free the higher-level control systems from unnecessary and calculation-intensive ballast. The system configuration is therefore accordingly simple, as the higher-level control system just outputs the actual target value and monitors some feedback information. The following sections will implicitly refer to *LinMot*® *P* linear motors as actuators when no specific indication is provided. In this case the target value consists in the position set value to which the drives should move, while the feedback information typically represents a following error message. In the case of stepper motors and electromagnets these statements can be interpreted by analogy.

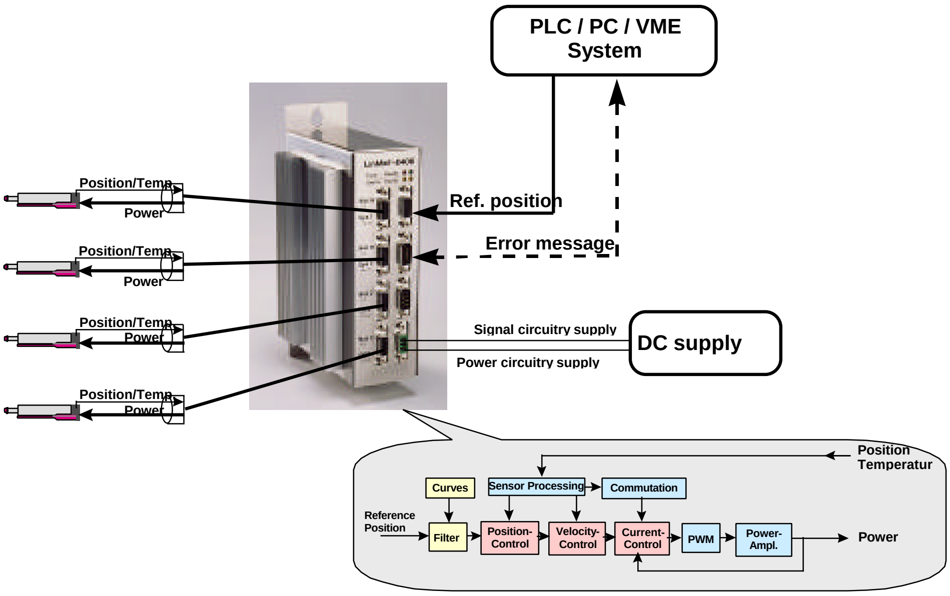


Figure 3-3: *LinMot*®'s peripheral position control frees the higher-level control system from calculation intensive tasks.

3.6.2 Operation modes

Different modes of operation are available to the user. These allow optimal embedding of the actuators in the control concept of the machine and the respective application. The *LinMot*® AT software currently supports the following modes of operation:

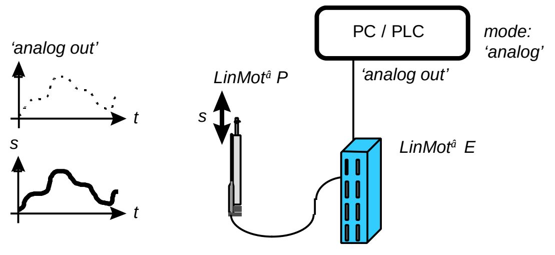
- direct digital reference value setting
- setting two target values depending on a high/low signal
- executing two predefined curves based on a trigger signal
- continuous operation, e.g. periodical execution of a curve

In all these modes a filter can be connected in order to limit the derived signals like for example the speed or acceleration. The following table shows typical application examples for the different modes of operation.

1. Example: Online specification of an exact reference curve

An higher-level control system outputs a reference curve online for the motor to follow exactly.

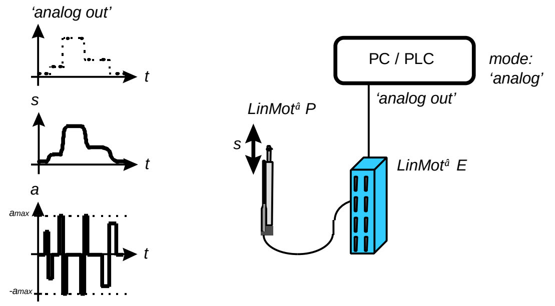
Solution:
Operating mode ‘analog’ without limiting filter settings.



2. Example: Online specification of any reference curve

An higher-level control system outputs position set values online for the motor to follow smoothly.

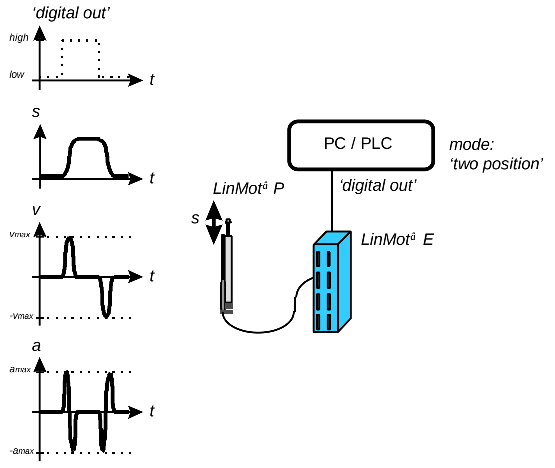
Solution:
Operating mode ‘analog’ with v_{max}/a_{max} filter activated.



3. Example: Digital selection between two set values

Following a digital high/low signal from a PLC, the motor moves between two given positions with presettable maximum acceleration and speed.

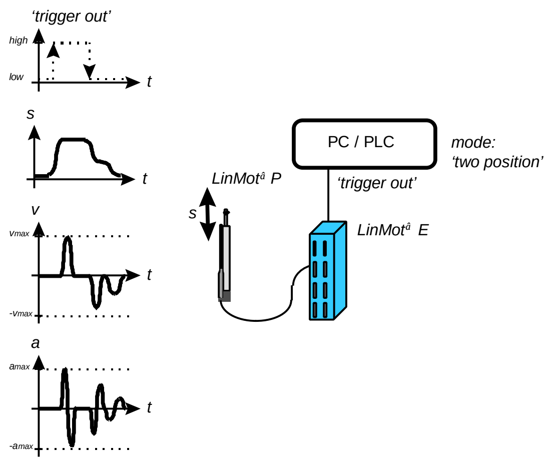
Solution:
Operating mode: ‘Two Point’. Acceleration limiter and speed limiter set to the desired maximal values.



4. Example: Digital selection between two reference curves

Following a trigger signal from a PLC, the motor moves between two given position following predefined trajectories.

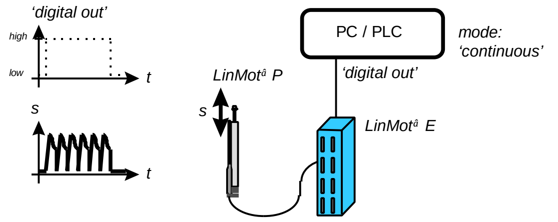
Solution:
Operating mode: ‘Trigger Curve’.
Acceleration limiter and speed limiter set to the desired maximal values.



5. Example: Periodical movement sequence

The motor executes a previously defined curve cyclically with a given frequency.

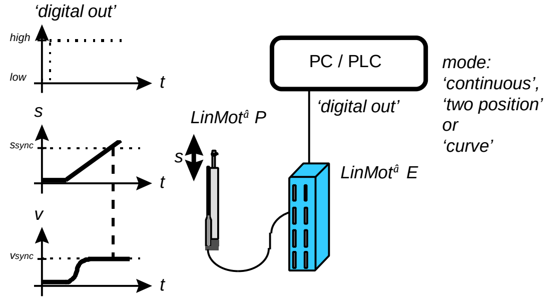
Solution:
Operating mode: ‘Continuous Curve’.
Acceleration limiter and speed limiter set to the desired maximal values.



6. Example: Synchronization on a speed value

A linear movement should be executed from a particular point and with a previously defined speed.

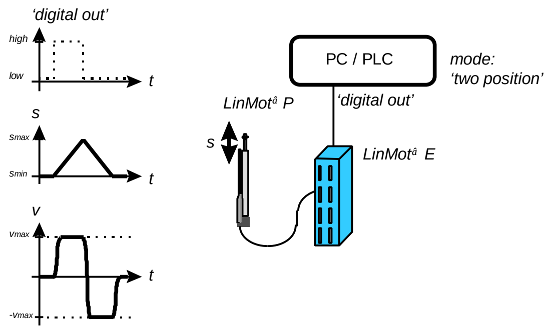
Solution:
Operating mode: ‘Analog’,’Two Position’ or ‘Trigger Curve’.
The reference position should be set in such a way, that the speed limiter will be active in the synchronization range.



7. Example: Movement with constant speed

A constant, linear movement should be triggered by means of a simple digital signal.

Solution:
operating mode: ‘Two Position’ with filter. Set the end positions to the boundaries of the stroke range and the v_{max} parameter to the desired maximum speed.
Hint: The movement can be stopped at any time with the FREEZE signal.



3.6.3 Connection to higher-level control systems

The electronic units *LinMot*® *E* can be controlled by higher-level control systems directly by means of analog or digital input/output signal lines. The necessary parameterisation is done with the *LinMot*® *Talk* software via an RS-232 connection. Networking via CAN-bus or RS-232 is also possible on specific customer request.

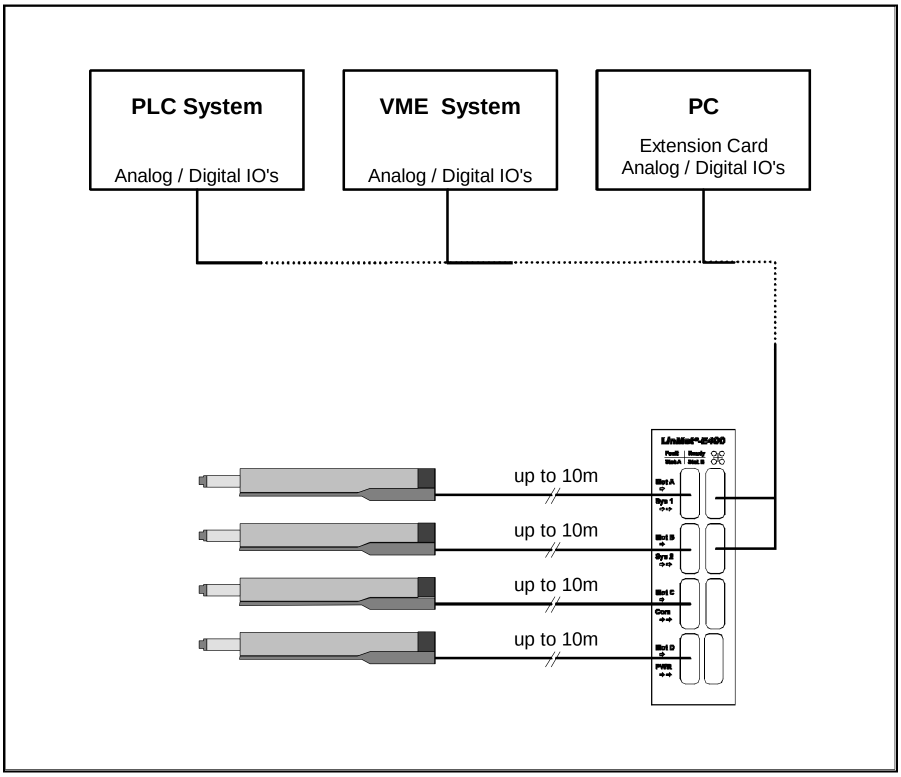


Figure 3-4: Connecting LinMot® to an higher-level control system

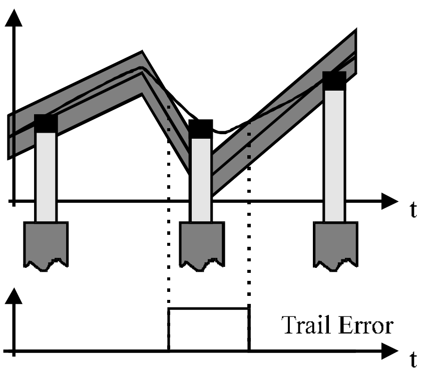
3.7 Protection and error behavior

3.7.1 Internal protection functions

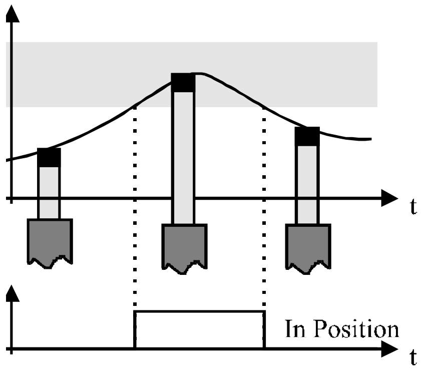
The integrated sensors and complex monitoring programs allow the detection and handling of thermal overloading of the *LinMot*® *E* and *P* components by means of warning and/or error messages. There is further the possibility to monitor the voltage supply of the *LinMot*® *E* electronic units for overvoltage and undervoltage. If the higher-level control system does not respond to the warning and error messages, the actuators will be shut down for safety reasons.

3.7.2 Monitoring functions

The so-called following error monitoring and the position range monitoring offer two powerful functions for the monitoring of the actual movement processes.



Following error monitoring



Position range monitoring

3.7.3 Emergency stop possibilities

From the point of view of the higher-level control system there are two ways to handle the *LinMot*® actuators in case of an emergency stop:

- With the so-called FREEZE signal the drives can be stopped and maintained in their current position. The movement is resumed when the FREEZE signal becomes inactive again.
- By means of the STOP signal the drives can be driven to a predefined emergency stop position or shut down.

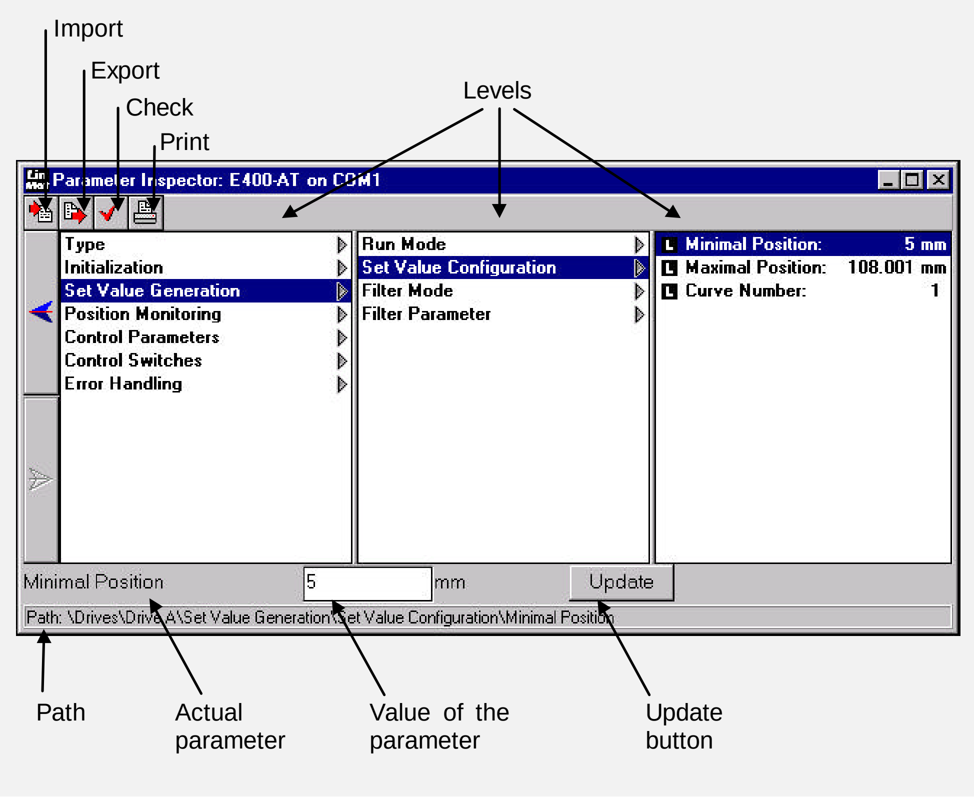
3.7.4 Emergency stop through supply interruption

As the *LinMot*® electronic units feature separate supplies for the signal circuitry and the power circuitry, an emergency stop can also be forced by directly interrupting the power supply to the power circuits. As long as the signal circuitry remains powered, the initialization and actual position of the *LinMot*® *P* actuators are not lost. Therefore it is possible to resume operation without needing to re-initialize the drives.

3.8 Operation and configuration

3.8.1 System configuration

The configuration and commissioning of the *LinMot*® systems is done by means of the MS-Windows based software *LinMot*® *Talk*. All the parameters can be displayed clearly, selected and edited with *LinMot*® *Talk* by simple clicking with the mouse. As all the parameters are stored in the electronic unit, they can be recalled, analyzed and changed at any time. Creating duplicates of parameters and parameter files to be archived is equally easy. Password protection protects the data from illegitimate access.



Minimal Position: 5 mm
Maximal Position: 108.001 mm
Curve Number: 1

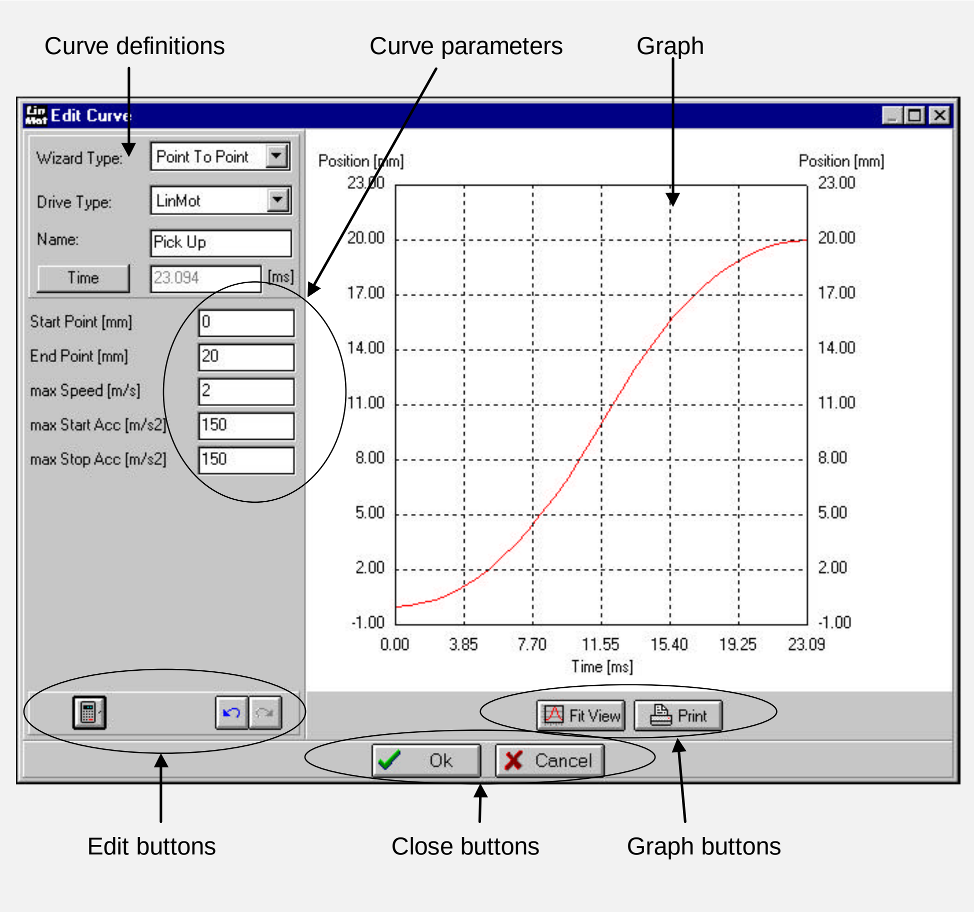
Minimal Position 5 mm

Path: \Drives\Drive A\Set Value Generation\Set Value Configuration\Minimal Position

Figure 3-5: Parameter Inspector window

3.8.2 Curve generation

Predefined reference curves can be executed on trigger signals in different operating modes. The shapes of these curves can be generated automatically by means of the Curve Creator, or can be edited manually point by point as Excel tables.



Wizard Type: Point To Point

Drive Type: LinMot

Name: Pick Up

Time 23.094 [ms]

Start Point [mm] 0

End Point [mm] 20

max Speed [m/s] 2

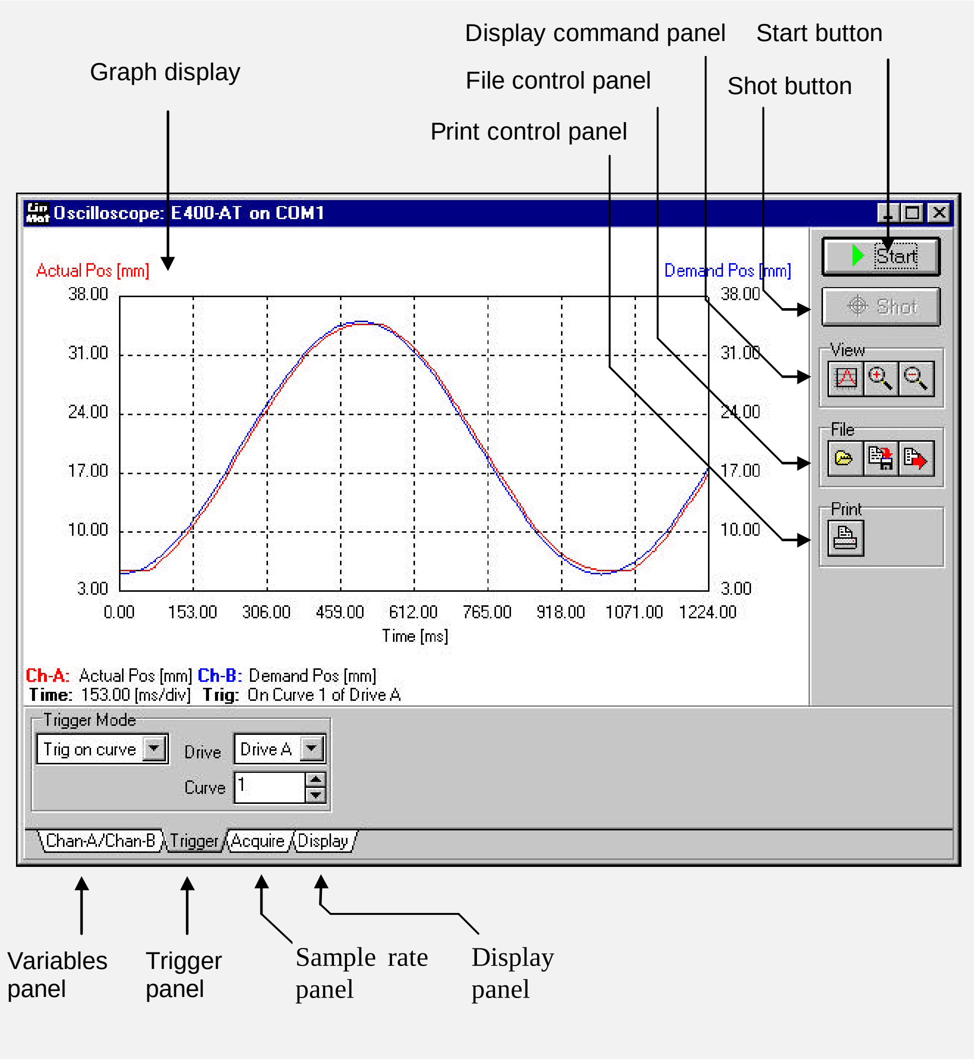
max Start Acc [m/s2] 150

max Stop Acc [m/s2] 150

Figure 3-6: Reference curves can be generated semi-automatically

3.8.3 Monitoring function (digital Oscilloscope)

Quick commissioning and optimization of a *LinMot*® system is supported by the built-in monitoring function. With it, the reference curves can be compared with the movements actually executed with no need for additional sensors, etc. The differences between the two curves can give indirect hints on counteracting forces and friction forces which can be useful when adjusting the controller parameters. The operation and look of the monitoring function correspond to those of modern digital oscilloscopes.



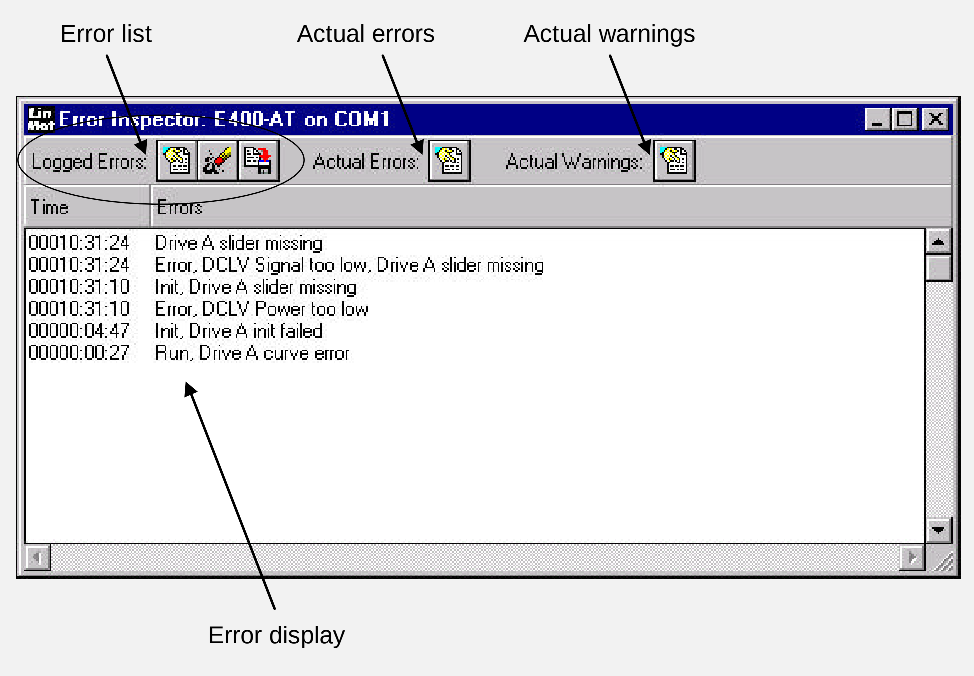
Ch-A: Actual Pos [mm] **Ch-B:** Demand Pos [mm]
Time: 153.00 [ms/div] **Trig:** On Curve 1 of Drive A

Trigger Mode: Trig on curve
Drive: Drive A
Curve: 1

Figure 3-7: Motion analysis with the integrated digital Oscilloscope

3.8.4 Error logbook

Thanks to the system clock built in the *LinMot*® *E* electronic units the errors that have occurred during operation can be logged on a time basis. The information is stored in the non volatile memory and allows a later reconstruction of the events.



```
00010:31:24   Drive A slider missing
00010:31:24   Error, DCLV Signal too low, Drive A slider missing
00010:31:10   Init, Drive A slider missing
00010:31:10   Error, DCLV Power too low
00000:04:47   Init, Drive A init failed
00000:00:27   Run, Drive A curve error
```

Figure 3-8: Error messages are saved to the logbook for later analysis.

3.9 Customer specific applications

The goal in the basic philosophy of *LinMot*® is to be able to cover the broadest range of applications simply by configuring the basic system, without special customer specific adjustments. In some cases (big series, special system environments) however customer specific adjustments can make sense. In these cases the necessary interventions have to be carried out by the manufacturer.
In most cases the customer specific changes will be realized in the software of the electronic unit ('Customized Application Software'). In most cases this will relate to the following two requirements:

- Additional control functions
- Networking with customer specific bus systems (CAN-Bus, RS-232)

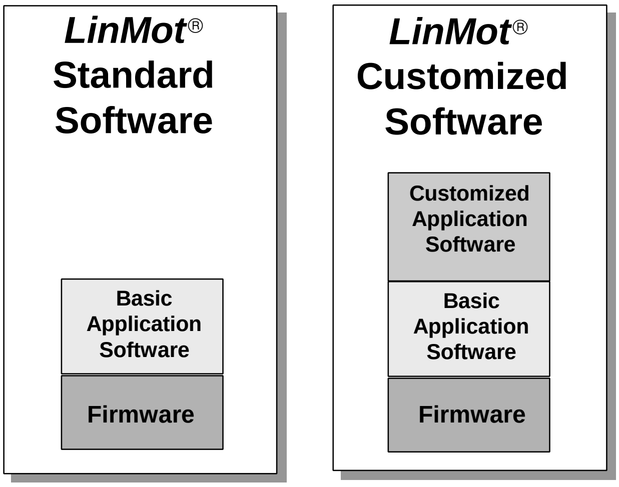


Figure 3-9: Structure of the *LinMot*® software: The manufacturer can realize special customizations in a special software segment ('Customized Application Software').

4. Design and Installation

This chapter discusses the different possibilities for driving the *LinMot*® products from a higher-level control system as well as the operating states of the control electronic unit in detail. Further, the interfaces and the installation are described in order to get familiar with the possibilities of the *LinMot*® products and be able to optimally embed them in the application.

This chapter shows the basic functions of the LinMot system explaining the –AT functionality as an introduction. For extended functionality of –MT and –DP electronic units, please read the manual for SW Rel. 1.3.

4.1 Operating modes

A wide range of operating modes are available for driving (setting the reference values) the actuators connected to the electronic unit. The desired set values can be given directly through an analog interface, or else they can be stored in the electronic unit in form of curves and tables, that can be run on a digital trigger signal from the higher-level control system.

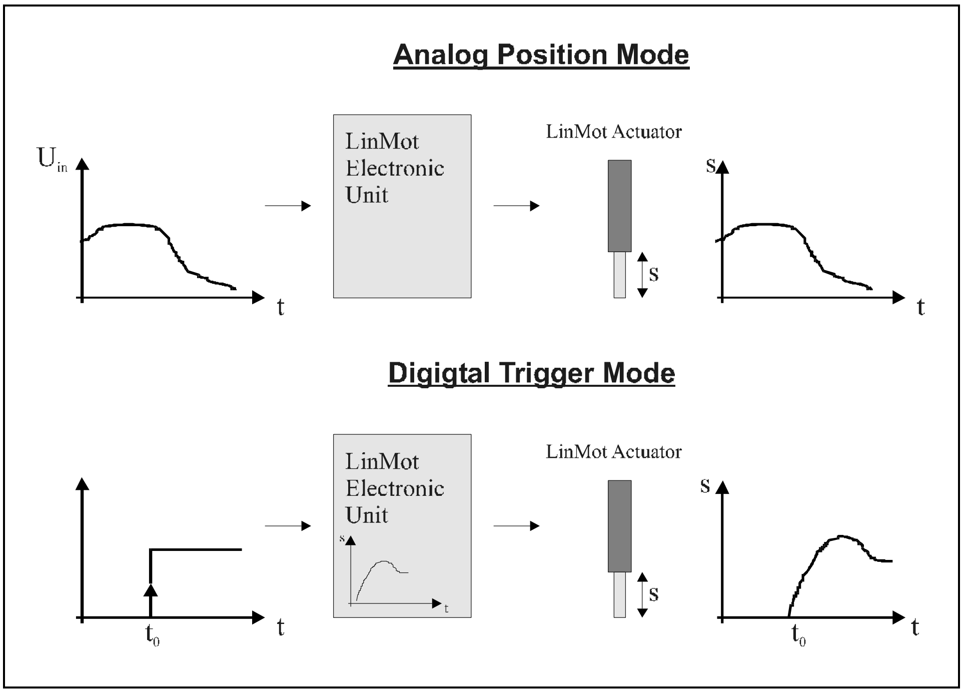


Figure 4-1: Operating modes of the LinMot electronic unit

A detailed description of the operating modes that can be configured for each connected motor individually is given in the following chapters. The operating modes differ according to the different actuators linear motor *LinMot*® *P*, stepper motor and inductive load (magnet, valve, ...) and are therefore treated separately.

4.1.1 Analog position setting

In the operating mode “Analog position setting” the desired position set value for the actuator is given by the higher-level control system directly by means of an analog input signal in the 0-10V range. The resolution of the internal A/D converter is 10 bit.

LinMot P

The desired position set value of the slider is given by the higher-level control system directly by means of an analog input signal. In the electronic unit a stroke range can be freely defined, that will correspond to the 0-10V range of the analog signal from the interface.

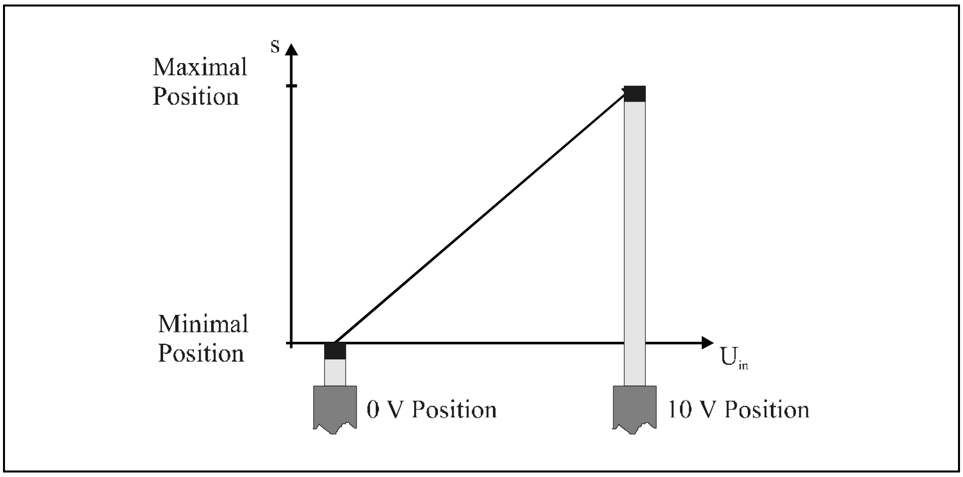


Figure 4-2: Analog position definition

In applications where the movement has to be executed with a controlled acceleration, the enormous acceleration capabilities of the linear drives of the LinMot series can be a disturbing factor. For this reason the maximal allowed acceleration and speed for each motor can be specified individually. This way the slider will accelerate within the limits of the application even on jumps of the reference position signal.

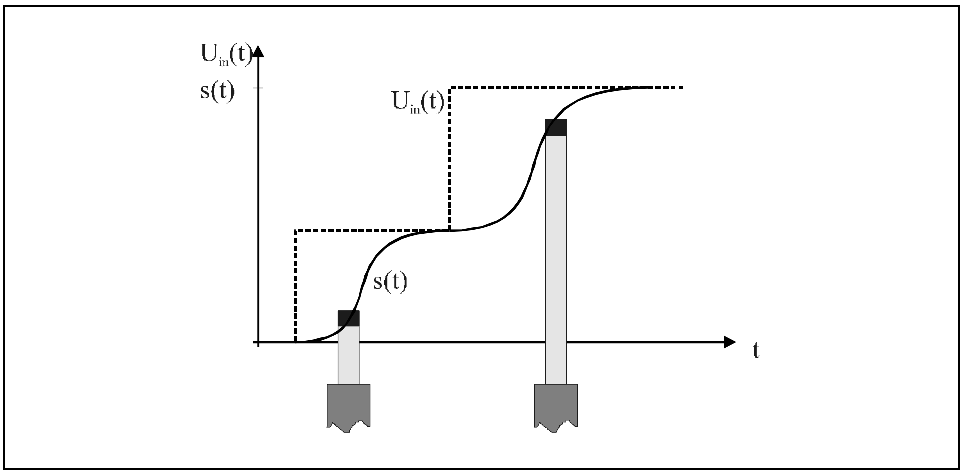


Figure 4-3: Limitation of the speed and acceleration

Stepper motor

The desired position set value of the rotor is given by the higher-level control system directly by means of an analog input signal. In the electronic unit a position range can be freely defined, that will correspond to the 0-10V range of the analog signal from the interface (the range can extend over more than one revolution).

To make sure that the stepper motor will not miss steps on very fast changes of the position set value, a maximal acceleration and a maximal rotational speed can be specified, that will not be exceeded in case of jumps in the reference position signal.

4.1.2 Analog current setting

The direct current setting over the analog interface can only be selected for an inductive load.

Inductive load

In the “Analog current setting” mode the desired output current to the inductive load is given by the higher-level control system directly through an analog input signal.

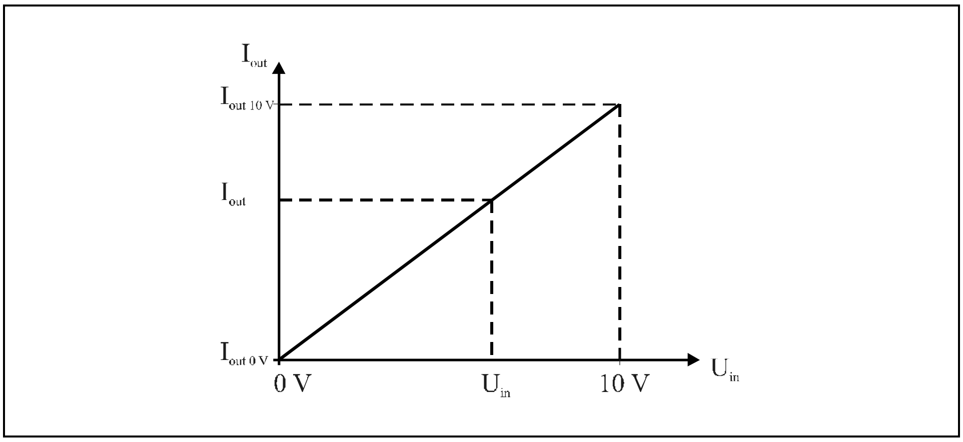


Figure 4-4: Analog current setting

In the electronic unit a current range can be freely defined, that will correspond to the 0-10V range of the analog signal from the interface.

4.1.3 Two point run

The operating mode “Two point trigger” is the simplest way to drive the actuators from the higher-level control system by means of digital trigger signals.

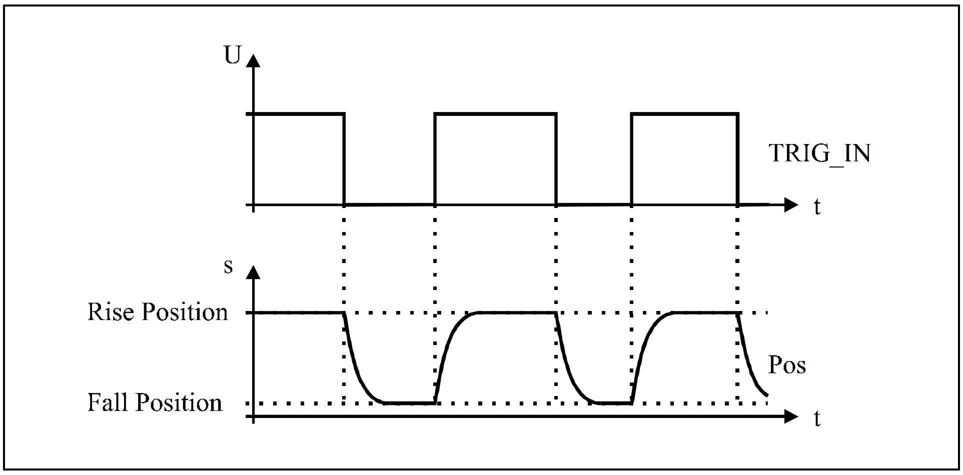


Figure 4-5: Two point run

The set values for the High level and the Low level of the input signal are stored separately in the electronic unit. On a change of the digital trigger signal on the electronic unit’s input, the corresponding set value is output.

LinMot P When using the linear motors of the *LinMot® P* series two position set values can be saved, to which the slider will move upon state changes of the input signal. Further, a maximal acceleration and a maximal speed can be specified in order to control the way the slider moves to the preset positions.

Stepper motor Two position set values can be saved for the stepper motor, too. To prevent the motor from missing steps the maximal acceleration and the maximal rotational speed at which the rotor will move can be specified.

Inductive load When driving an inductive load two current set values can be defined, that will be output depending on the input signal state.

4.1.4 Running reference curves

In the operating mode "Run reference curve" the actuators are driven by the higher-level control system by means of trigger signals.
In the electronic unit two reference curves are stored for each actuator: one for the Low-High state change and one for the High-Low state change of the trigger signal. When the trigger signal state changes at the input of the electronic unit, the corresponding curve is run. The electronic unit can store curves with up to 4'000 reference positions.

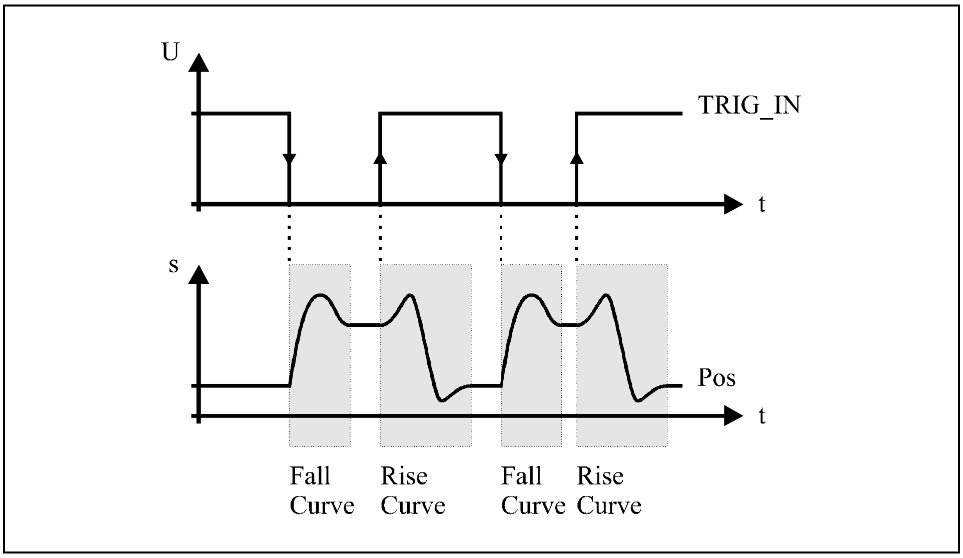


Figure 4-6: Running reference curves

LinMot P When using the linear motors of the LinMot P series two reference curves can be saved for each motor. These will be run after each corresponding state change of the trigger signal. The slider then maintains the last position of the curve until the next trigger starts the other curve.

Stepper motor Two reference curves can be saved for stepper motors, too. These will be run after each corresponding state change of the trigger signal.

Inductive load When driving an inductive load two reference curves can be defined for the current that will be output depending on the trigger signal state.

4.1.5 Set values through the serial interface

The following serial hardware interfaces are available:

RS232

For standard RS232 serial protocol and commands, please refer to the SW Rel. 1.3 manual

RS485

For standard RS485 serial protocol and commands, please refer to the SW Rel. 1.3 manual.

CAN BUS

Can Bus interface (Normal CAN version with 11 Bit identifier and Extended CAN version with 29 Bit identifier) with transfer rates up to 1 Mbaud.
For special applications Sulzer Electronics AG offers the possibility to implement a specific CAN protocol for the transmission of the set values in the electronic unit.

Sulzer Electronics AG will be happy to give any further information that may be needed regarding the implementation of specific protocols.

4.2 Operating states

The operating modes described in the previous chapter have shown how the actuators can be driven when the electronic unit is on (i.e. when it is in the state RUN). This chapter gives a description of all the operating states.

Overview

The following are the possible operating states of the electronic unit:

SETUP	The system is being started
WAIT FOR DISABLE	Prevents uncontrolled starting
DISABLE	The system is ready to run
DRIVE INIT	Initialization of the drives
RUN	The motors run
STOP	Stop behavior
ERROR	Error state

The following figure shows the states and the possible state transitions.

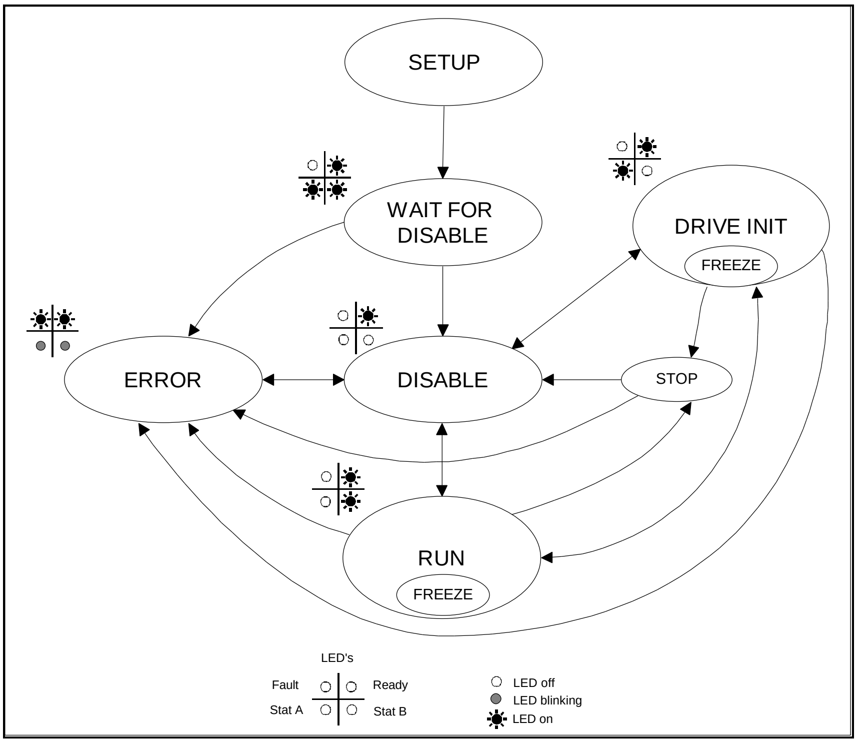


Figure 4-7: Operating states

Control signals

A state change can either be caused by means of the digital control signals to the electronic unit or, in certain cases by the electronic unit itself (e.g. in case an error occurs).

The following are the control signals of the electronic unit:

INIT	Initialization of the drives
RUN	The actuators are powered
FREEZE	The actuators maintain the actual position
STOP	Change to the operating state Stop

Status display

The actual operating mode can be read at all times from the four status LEDs on the front of the electronic unit.

READY	The system has started correctly
FAULT	An error has occurred
STAT A	Coding for the actual operating state
STAT B	Coding for the actual operating state

The display of the various operating states is shown in the figure on the previous page. In the state ERROR a blink code of the STAT LEDs tells what the actual error is. The blink codes are explained in chapter ‘Service’.

4.2.1 Operating state “Setup”

As soon as the electronic unit is powered on it goes to the state SETUP, while the system is initialized and all parameters are loaded. A check is also done to make sure that all the software modules needed for the correct operation of the unit have been loaded.

Next operating state

- WAIT FOR DISABLE: the transition occurs automatically after SETUP.

4.2.2 Operating state “Wait for Disable”

The operating state WAIT FOR DISABLE, in which all actuators are off, prevents the system from starting uncontrolled upon startup. A change to the state DISABLE occurs only when the RUN and INIT signals are low (0V). This security feature can be disabled in the Auto-Start Mode, where the system starts automatically.

Next operating state

- DISABLE: if the INIT and RUN inputs are not active.
- DISABLE: the transition is automatic in the Auto-Start Mode.
- ERROR: the transition occurs unconditionally when an error occurs.

4.2.3 Operating state “Disable”

The actuators remain unpowered in this state. The system to switch to the states RUN or DRIVE INIT.

Next operating state

- DRIVE INIT: if the INIT input becomes active.
- RUN: if the RUN input becomes active and all the drives are already installed.
- DRIVE INIT: the transition occurs automatically in the Auto-Start Mode.
- ERROR: the transition occurs unconditionally when an error occurs.

4.2.4 Operating state “Drive Init”

This operating state serves the zero-position-initialization of those actuators that don’t come with an absolute measuring system. The actuators to be initialized are powered and controlled. The different initialization possibilities will be described in detail in chapter 4.4.

Before all the motors have initialized a warning is active, that prevents the system from jumping to the RUN state. The system must go through this state at least once.

Freie

FREEZE is a substate of the DRIVE INIT state that can be reached by means of the Freeze control signal. In the substate FREEZE all actuators remain powered and controlled in their respective actual positions. After resetting the Freeze control signal the system returns to the state DRIVE INIT unchanged.

Next operating state

- DISABLE: if the INIT input becomes inactive and the RUN input is inactive.
- RUN: if the RUN input is active and the INIT input becomes inactive (in case the drives are already installed).
- RUN: this transition occurs automatically in the Auto-Start Mode as soon as the initialization is completed.
- STOP: if the STOP signal becomes active.
- ERROR: the transition occurs unconditionally when an error occurs.

4.2.5 Operating state “Run”

In the state RUN, the actual working state, the actuators are powered and controlled.

Freeze

FREEZE is a substate of the RUN state that can be reached by means of the Freeze control signal. In the substate FREEZE all actuators remain powered and controlled in their respective actual positions. After resetting the Freeze control signal the system returns to the state DRIVE INIT unchanged.

Next operating state

- DISABLE: if the RUN input becomes inactive and the INIT input is inactive.
- DRIVE INIT: if the RUN input becomes inactive and INIT is active.
- STOP: if the STOP signal becomes active.
- ERROR: the transition occurs unconditionally when an error occurs.

4.2.6 Operating state “Stop”

A transition to this state takes place when the Stop signal becomes active, which means the higher-level control system requested an emergency stop. In this state the actuators can be either turned off, freezed in the actual position or driven to an emergency stop position according to their respective configuration.

Next operating state

- DISABLE: if the inputs RUN, INIT and STOP are inactive.
- ERROR: the transition occurs unconditionally when an error occurs.

4.2.7 Operating state “Error”

Upon an error a transition to this state takes place unconditionally and from any state the system might be in at that moment. The motors are not powered and the state can only be left when the error has been fixed or removed. The actual error is displayed by means of a blink code and all errors are stored in the electronic unit and can be read by means of the PC-software.

Next operating state

- DISABLE: if the error has been removed and the RUN and INIT inputs are inactive.

4.3 Position monitoring

The different operating modes for driving the electronic unit from a higher-level control system have been shown in chapter 4.1. In almost all application cases the higher-level control system will need additional information and feedback about the actual positions of the driven actuators.

For example, the higher-level control system might need to be informed about whether the desired movement sequence was executed, or if all actuators are already outside the motion range of another moving machine part, so that the latter can safely move without damaging the other actuators.

Two digital signals are available for this feedback to the higher-level control system. They allow the monitoring of the following error and of a position range. These messages are forwarded as a collective message (or concatenation) from all the actuators connected to an electronic unit.

4.3.1 Following error monitoring

For the following error monitoring the difference between the position set value and the actual position is monitored. If this difference becomes bigger than the maximal allowed position difference that was set, it is to assume that a disturbance of some kind has occurred (e.g. slider jammed, load mass too big, etc.) which has to be communicated to the higher-level control. In this case a warning message or an error message should be sent to the higher-level control system.

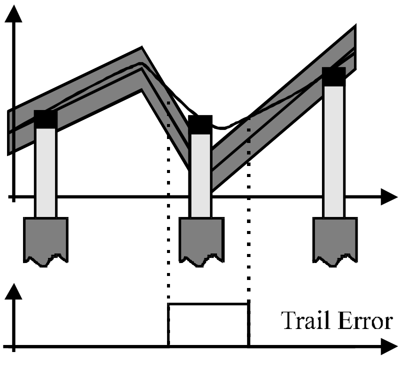


Figure 4-8: Following error monitoring

Configuration possibilities

A positive and a negative maximal allowed position difference can be set for each actuator. Here it can be freely specified whether the event of trespassing these limits should generate a warning message or an error message (turning off the actuator) to be sent to the higher-level control.

4.3.2 Position range monitoring

In cases where two actuators share a common motion range, the higher-level control must be sure the first actuator is outside the motion range of the second one before moving the latter to it. Therefore some feedback is needed when all actuators are within a safe range.

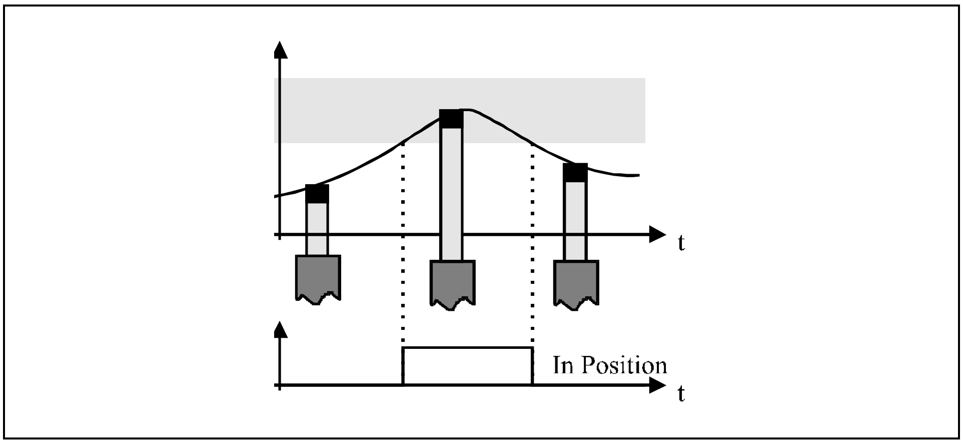


Figure 4-9: Position range monitoring

Configuration possibilities

A position range can be specified for each actuator. As soon as an actuator's actual position is outside this range a message is sent to the higher-level control system.

4.4 Initialization

Upon powering on the electronic unit all actuators that operate without an absolute measuring system (*LinMot*® *P*, stepper motors) must be initialized. The initialization is done in form of a reference run, by which the zero-position of the actuator is determined.

After powering on the electronic unit a warning message is sent over the WARN output to indicate that the motors have not yet been initialized. The initialization procedure always starts from motor A. As soon as this is done the initialization of motor B begins, and so on. After motor D has initialized the warning message is removed.

4.4.1 Linear motors of the *LinMotâ P* series

The linear motors of the *LinMot*® *P* series must be initialized after the powerup of the electronic unit. After the initialization the actual position of the slider is stored in the electronic unit until the signal power supply to the unit is interrupted. The actual position is still maintained when the linear motor is not powered and the slider is moved from its position.

Three types of initialization are available for the linear motors of the *LinMot*® *P* series:

Initialization to a stop

When initializing, the slider is moved with a user-definable speed until it encounters a stop. As soon as the slider stops its position is taken as the zero-position.

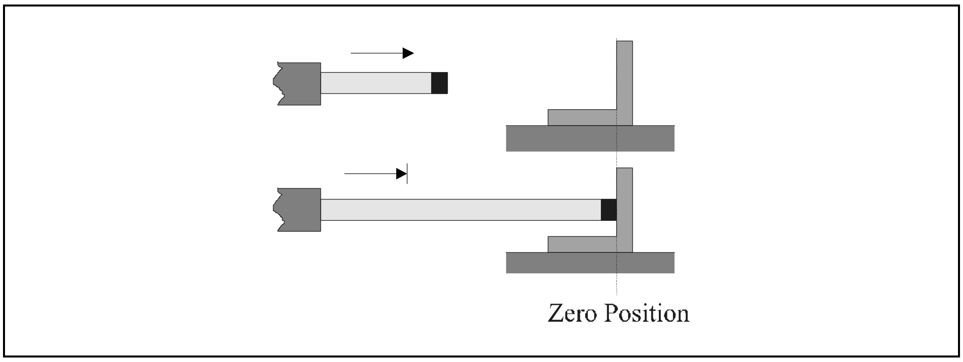


Figure 4-10: Initialization to a stop (move out)

In this mode it can be specified whether the slider should be moved outwards or inwards (the figure shows the “move out” option).

Hint: by using a stop that is fixed to the machine and not to the stator of the linear motor, re-adjusting the stator position will not be necessary when replacing the motor, because the reference stop for the zero position remains the same.

Initialization on a trigger

In order to initialize the motor by means of a trigger signal a digital position sensor is needed to detect the slider when it is in the desired zero-position. The slider moves with a user-definable speed over the position sensor. As soon as the sensor detects the slider the actual position is taken as the zero-position.

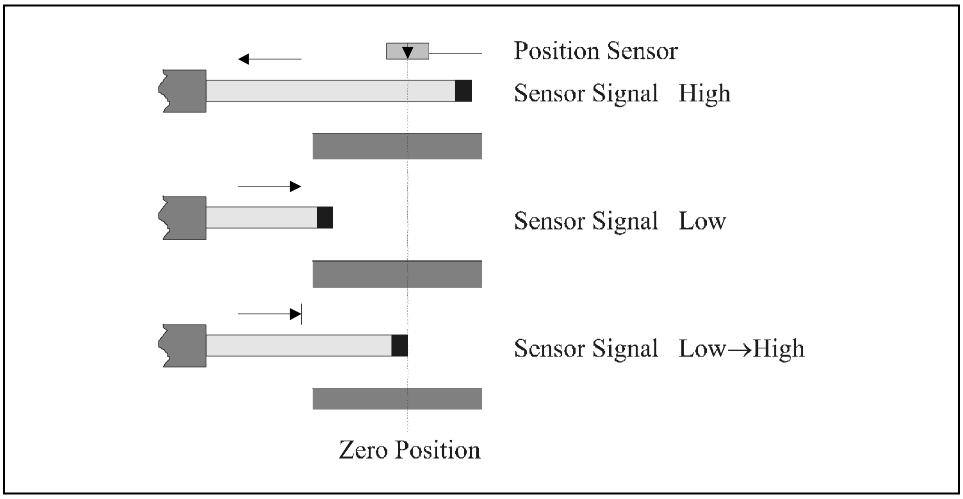


Figure 4-11: Initialization with a position sensor

The trigger signal of the sensor is forwarded to the electronic unit by the higher-level control system. Here, too, it can be specified whether the system should consider the active edge of the trigger signal when the slider is moved inwards or outwards.
If the trigger signal is already High when the initialization is started, the slider is first moved in the opposite direction until the signal goes low. At this point the slider is moved in the specified direction to trigger the active edge.

Initialization on actual position

In this third case no reference run is executed. The actual position of the slider upon powerup of the electronic unit is saved as zero-position. This implies that the slider must already find itself in the desired zero-position before powering up the unit.

Testing for free stroke range

After the initialization is done in one of the three modes it is possible to check if the slider can move freely along its entire stroke range. During this process the slider is moved over the specified range with same specified speed. Should the slider encounter an obstacle within its stroke range an error message is sent out to inform that the initialization was not completed successfully.

4.4.2 Stepper motors

Stepper motors have to be initialized every time they are turned on. As they can only be operated without a pick-up, their actual position cannot be detected when the rotor is moved in a non-powered mode (e.g. the operating state Disable).

Stepper motors can be initialized in two different ways:

Rotate until trigger

In order to initialize the motor by means of a trigger signal a digital position sensor is needed to detect the rotor when it is in the desired zero-position. The rotor rotates in the specified direction with a user-definable speed over the position sensor. As soon as the sensor detects the rotor the actual position is saved as the zero-position.

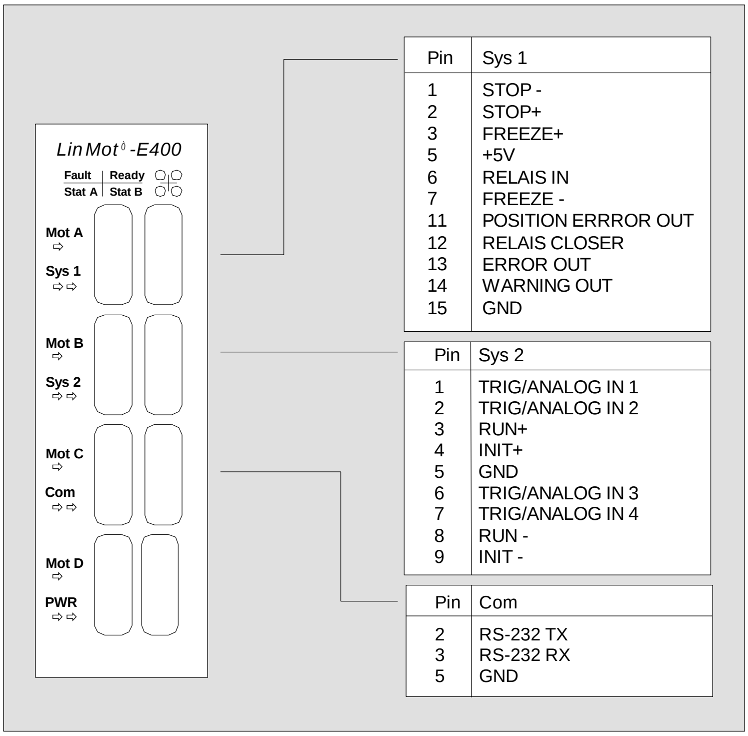
If the trigger signal is already High when the initialization is started, the rotor is first moved in the opposite direction until the signal goes low. At this point it is moved in the specified direction to trigger the active edge.

Initialization on actual position

In this case no reference run is executed. The actual position of the rotor upon powerup of the electronic unit is saved as zero-position. This implies that the rotor must find itself in the desired zero-position already before powering up the unit.

4.5 Signal interfaces

The signal interfaces of the electronic unit families E100/E200/E400 and E1000/E2000/E4000 are identical and are described together in this chapter.



Pin	Sys 1
1	STOP -
2	STOP+
3	FREEZE+
5	+5V
6	RELAIS IN
7	FREEZE -
11	POSITION ERRROR OUT
12	RELAIS CLOSER
13	ERROR OUT
14	WARNING OUT
15	GND

Pin	Sys 2
1	TRIG/ANALOG IN 1
2	TRIG/ANALOG IN 2
3	RUN+
4	INIT+
5	GND
6	TRIG/ANALOG IN 3
7	TRIG/ANALOG IN 4
8	RUN -
9	INIT -

Pin	Com
2	RS-232 TX
3	RS-232 RX
5	GND

Figure 4-12: Signal interfaces

The interfaces are accessible from the front of the unit through three D-Sub connectors SYS1, SYS2 and COM.

4.5.1 SYS1 interface

The operating states FREEZE and STOP of the electronic unit are driven through the inputs of the SYS1 interface. The outputs WARNING OUT, ERROR OUT and POSITION ERROR OUT carry internal warnings and error messages to the higher-level control system. Further, a message output MSG for internal state output is available.

Pin	Description	Pin	Description
1	STOP -	9	Do not connect
2	STOP +	10	Do not connect
3	FREEZE +	11	POSITION ERROR OUT
4	Do not connect	12	MSG
5	+5V	13	ERROR OUT
6	Do not connect	14	WARNING OUT
7	FREEZE -	15	GND
8	Do not connect		

Table 4-1: Pin configuration of the SYS1 connector

STOP

Starts the “Emergency Stop” behavior that was specified. Here it can specified whether the slider should be stopped with the maximal allowed acceleration, taken to a predefined emergency stop position or the motor should be turned off.

Data: Galvanically separated digital input (Low level active)
Input voltage: 0…24V DC (max. -10…26V DC)
For signal = 0 < 2V DC
For signal = 1 > 3.5V DC
Input current: < 20mA (24V)
Input delay: 1.6ms

FREEZE

The sliders of the linear motors are stopped with the maximal allowed acceleration.

Data: Galvanically separated digital input (High level active)
Input voltage: 0…24V DC (max. -10…26V DC)
For signal =0 < 2V DC
For signal =1 > 3.5V DC
Input current: < 20mA (24V)
input delay: 1.6ms

Digital outputs

The digital outputs are realized as open collector outputs and must be driven over an external pull-up resistor.

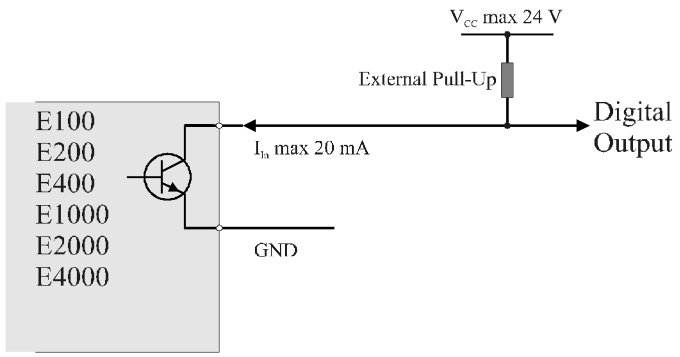


Figure 4-13: Digital output layout

During normal operation (no warnings or error messages) the digital outputs are pulled down to ground (GND) by the electronic unit. When an error or a warning is generated (or the signal cable is interrupted) the output becomes high impedance and is pulled up over the external resistor.

WARNING OUT

This is set when a warning (to be considered) is pending. A warning is to be considered as a precursor of an error.

Data: Open collector output (High level active)
Max. 24V / 20mA

ERROR OUT

This output is set when an error has occurred, that should be taken care of.

Data: Open collector output (High level active)
Max. 24V / 20mA

POSITION ERROR OUT

This output is set when the slider is outside the position range that was defined for it.

Data: Open collector output (High level active)
Max. 24V / 20mA

MSG Message output can be defined via the configuration software to suit your needs

+5V / GND 5V supply (digital). Output with a maximal load capability of 50mA

Caution: the pins 4,8,9,10 of Sys1 are exclusively for debugging purposes and must not be connected.

4.5.2 SYS2 connector

The operating states RUN and INIT of the electronic unit are driven through the inputs of the SYS2 interface. The set values are also given through this port.

Pin	Description	Pin	Description
1	TRIG / ANALOG IN A	6	TRIG / ANALOG IN C
2	TRIG / ANALOG IN B	7	TRIG / ANALOG IN D
3	RUN +	8	RUN -
4	INIT +	9	INIT -
5	GND		

Table 4-2: Pin configuration of the SYS2 connector

RUN During normal operation the actuators are turned on and off by means of the RUN signal.

Data: Galvanically separated input (High level active)
Input voltage: 0…24V DC (max. -10…26V DC)
For signal =0 < 2V DC
For signal =1 > 3.5V DC
Input current: < 20mA (24V)
Input delay: 1.6ms

INIT The initialization of the motors is started.

Data: Galvanically separated input (High level active)
Input voltage: 0…24V DC (max. -10…26V DC)
For signal =0 < 2V DC
For signal =1 > 3.5V DC
Input current: < 20mA (24V)
Input delay: 1.6ms

TRIG/ANALOG IN The inputs TRIG/ANALOG IN A, TRIG/ANALOG IN B, TRIG/ANALOG IN C and TRIG/ANALOG IN A are used in the states INIT and RUN. In the state RUN these inputs serve either the direct position value setting with analog signals or the triggering of reference curves that are stored in the electronic unit. When initializing with a position sensor, an active edge on this input defines the zero-position.

Data: Analog 0…10V / 100kΩ (10Bit resolution)
Digital max. 24V (Rin 100kΩ)

4.5.3 COM connector

This port serves the connection to the serial interface of a PC for the configuration with the *LinMot*® *Talk* software.

Pin	Description	Pin	Description
1	NC	6	NC
2	RS-232 TX	7	NC
3	RS-232 RX	8	NC
4	NC	9	NC
5	GND		

Table 4-3: Pin configuration of the COM connector

A 9-pole interface cable with 1:1 connections is needed for the connection of the electronic unit to a PC.

4.6 Actuator interfaces

The interfaces for the connection of the actuators are found on the front of the electronic unit and are named Mot A, Mot B, Mot C and Mot D.

Caution: The pin configuration for the actuators of the two electronic unit series E100/E200/E400 and E1000/E2000/E4000 are not identical. The incorrect connection of an actuator can lead to damage of the electronic unit and of the actuator itself.

4.6.1 Connection of *LinMot*â *P* linear motors

The sensor signals and the power signals of the linear motors of the *LinMot*® *P* series are carried on a single cable.

By default the *LinMot*® drives come factory-fitted with a cable. In cases where a longer cable is necessary, the cable can be extended up to 50m (see data book). Only the special LinMot cable (Art. No. 0150-1920) must be used for extension cables.

E100/E200/E400

9-pole D-Sub connectors are available on the front of the electronic units of the E100/E200/E400 series for connecting the actuators.

Pin	Color	Signal
1	red	Phase 1 +
2	blue	Phase 2 +
3	white	+5V
4	yellow	Sensor Sin
5	black	Temperature Sensor
6	pink	Phase 1 -
7	gray	Phase 2 -
8	brown	Ground
9	green	Sensor Cos

Table 4-4: E100/E200/E400 pin configuration for the *LinMot*â *P*

Only the special LinMot cable (Art. No. 0150-1920) must be used for extension cables. In applications where the motor cable has to be or moved during operation, the special trailing chain cable (Art. No. 0150.1927) must be used (max. 10m).

E1000/E2000/E4000

10-pole Mini-Combicon connectors are available on the front of the electronic units of the E1000/E2000/E4000 series for connecting the actuators.

Pin	Color	Signal
1	red	Phase 1 +
2	pink	Phase 1 -
3	blue	Phase 2 +
4	gray	Phase 2 -
5	white	+5V
6	brown	Ground
7	yellow	Sensor Sin
8	green	Sensor Cos
9	black	Temperature sensor
10		Shield

Table 4-5: E1000/E2000/E4000 pin configuration for the LinMot-P

The lower bit of the motor connector has the pin number 1 (with the unit upright). The pins are numbered from bottom to top.

Only the special LinMot cable (Art. No. 0150-1920) must be used for extension cables. In applications where the motor cable has to be or moved during operation, the special trailing chain cable (Art. No. 0150.1927) must be used (max. 10m).

4.6.2 Connecting stepper motors

As stepper motors can only be driven, not controlled, only the two phases need to be connected to the electronic unit.

E100/E200/E400

When driven by an electronic unit of the E100/E200/E400 family, stepper motors must be connected as shown in the following figure.

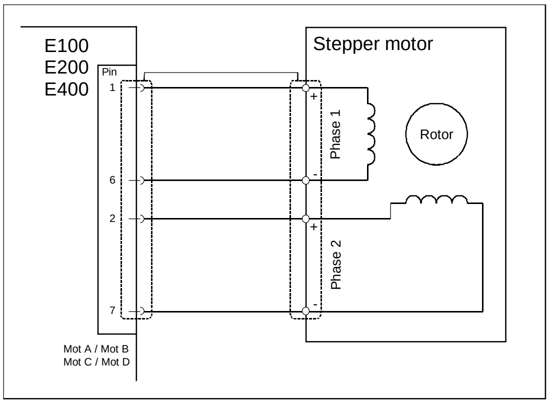


Figure 4-14: E100/E200/E400 Connection layout for stepper motors

Only shielded cable with a core cross section area of 0.20mm² should be used.

E1000/E2000/E4000

When driven by an electronic unit of the E1000/E2000/E4000 family, stepper motors must be connected as shown in the following figure.

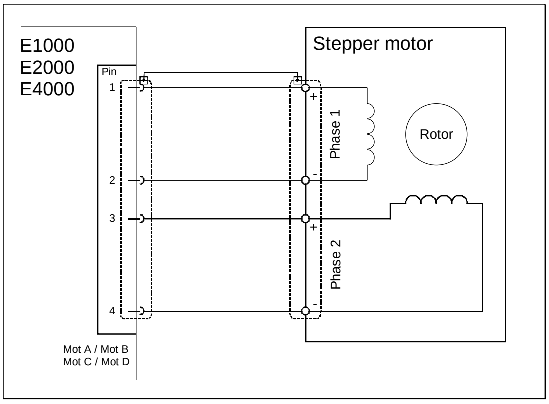


Figure 4-15: E1000/E2000/E4000 Connection layout for stepper motors

Only shielded cable with a core cross section area of 0.50mm² should be used.

4.6.3 Connecting inductive loads

Inductive loads can be connected to phase 1 of the electronic unit. The connection layout corresponds to the one for phase 1 of a stepper motor, as shown in the previous section.

4.7 Connecting the power supply

The supplies for the signal circuitry and for the power circuitry are carried out separately. For machines where the actuators have to be turned off in case of an emergency stop, the linear motors can be shut down simply by interrupting the supply to the power circuitry. The position data is maintained as long as the signal circuitry is powered.
The supply of both supply voltages should be realized as low ohmic and as low inductive as possible, to guarantee a stable and disturbance free power supply.

Caution: Exceeding the maximal allowed supply voltage for both the signal and the power circuitry inevitably damages the electronic unit. Only tested power supplies featuring potential separation must be used.

4.7.1 E100/E200/E400 supply

In the electronic units of the E100/E200/E400 series both signal and power circuitry are supplied over the 3-pole POWER SUBCON connector on the front. As the devices can be supplied with the same voltage for the power and the signal circuitry, a single supply can be used.

In order to guarantee safe operation, the electronic units must be connected according to the following figure.

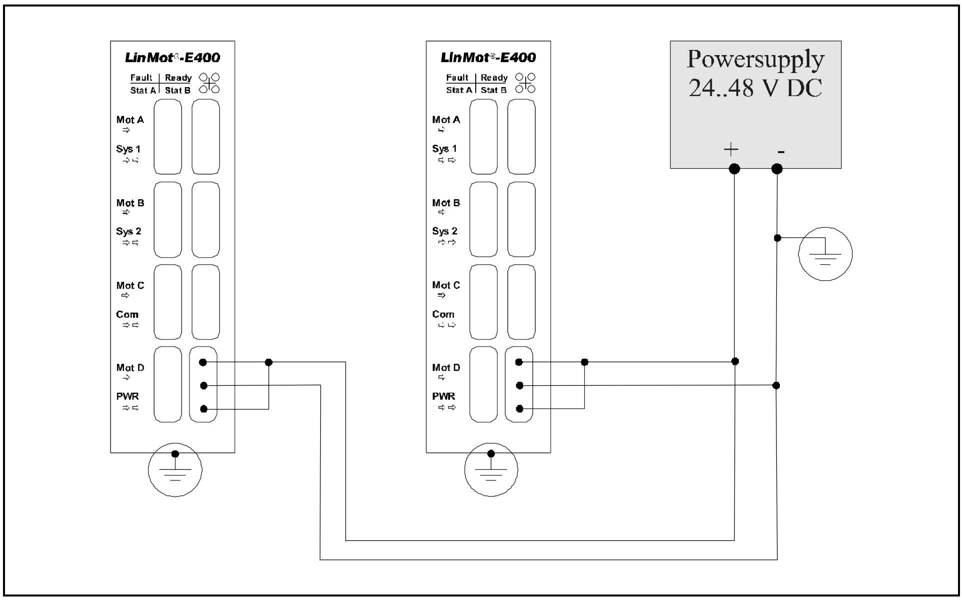


Figure 4-16: Installation of the E100/E200/E400 units

Caution: Before putting into operation, the housing of the electronic unit must be grounded. The grounds of the signal and power supplies must also be connected to earth at some point.

Pin	Signal
top	+Vcc Signal supply
middle	GND
bottom	+Vcc Power supply

Table 4-6: Supply pin configuration E100/E200/E400

Signals supply

Input voltage:	24 - 48V DC
Max. input current:	160 mA (at 24 V)

Power supply

The linear motors reach their maximal force and full dynamics only when supplied with a 48V DC power supply (see data sheets of the linear motors).
Input voltage: 24 - 48V DC

The required power strongly depends on the loading of the motors. For an optimal design of the power supplies the power requirements of each application should be clarified individually.

Practical applications have shown that in most cases, even when driving four linear motors *LinMot*® *P* at the same time with one E400 electronic unit a power supply of 48V/300W is sufficient.

4.7.2 E1000/E2000/E4000 supply

In the electronic units of the E1000/E2000/E4000 series the signal circuitry and the power circuitry are supplied over two separate connectors that are found on the front of the unit. As the supply voltage ranges for the two circuits are different, the only case in which they can be supplied from a single power supply is when they are operated at 48V.

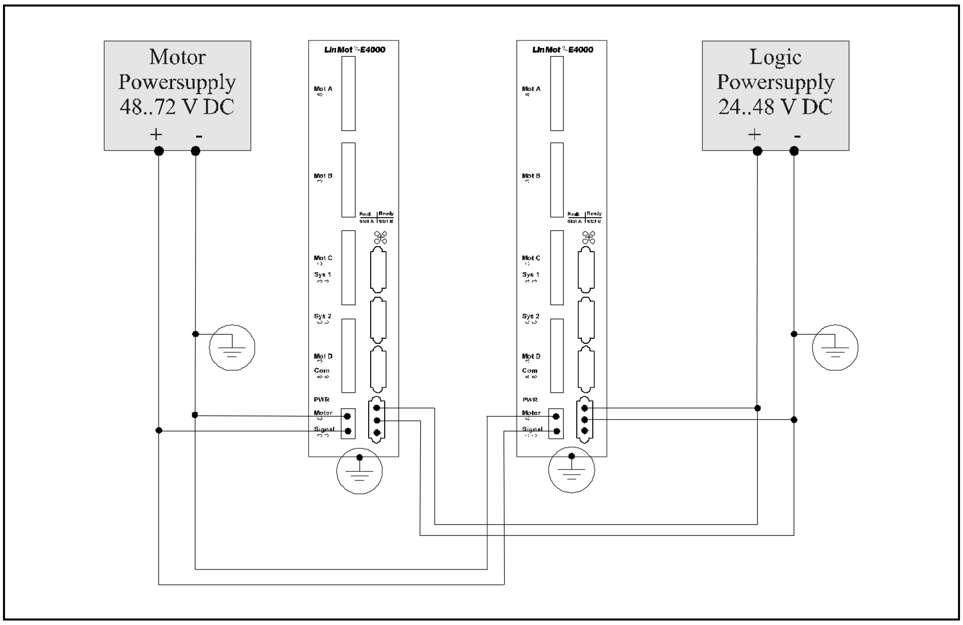


Figure 4-17: Installation of the E1000/E2000/E4000 electronic unit

In order to guarantee safe operation, the electronic units must be connected according to the figure.

Caution: Before putting into operation, the housing of the electronic unit must be grounded. The grounds of the signal and power supplies must also be connected to earth at some point.

Signal supply

Pin	Signal
top	+Vcc Power Supply Logic
middle	GND
bottom	do not connect !

Table 4-7: Signal supply pin connection for the E1000/E2000/E4000

Data: Input voltage: 24 - 48V DC
Input current: 400 mA (at 24 V)

Power supply

The linear motors reach their maximal force and full dynamics only when supplied with a 72V DC power supply (see data sheets of the linear motors).

Pin	Signal
top	GND
bottom	+Vcc Power supply

Table 4-8: Power supply pin configuration E1000/E2000/E4000

Data: Input voltage: 48 - 72V DC

The required power strongly depends on the loading of the motors. For an optimal design of the power supplies the power requirements of each application should be clarified individually.

Practical applications have shown that in most cases, even when driving four linear motors *LinMot*® *P* at the same time with one E4000 electronic unit a power supply of 72V/600W is sufficient.

4.8 HW configuration of the electronic units

The electronic units consist of a signal board and a power board that you can configure to suit your own requirements.

To configure the boards you must remove the heat sink and open the case. This is done by removing all the screws. The two boards are kept in place by spacers and connected by means of post connectors. When opening the unit and handling the electronic boards the usual care should be taken in observing the necessary measures to prevent electrostatic discharge that would otherwise damage the boards (ESD-mats, ground connection, ..).
Further care must be taken in order not to damage or stress either mechanical or electrical components, as this could lead to malfunction or damage.

Caution: The electronic unit must be completely assembled and closed in its case before it can be put back into operation.

4.8.1 Signal board configuration

The two electronic unit series E100/E200/E400 and E1000/E2000/E4000 differ only in the execution of the power board. The signal boards are absolutely identical and for this reason the explanations in this section are valid for both.

The interfaces to the higher-level control system can be configured on the signal board. Only the following jumper positions result in usable configurations of the electronic unit.

SYS1 interface

Instead of the digital open-collector outputs, relay closers or openers can be configured to act as outputs.

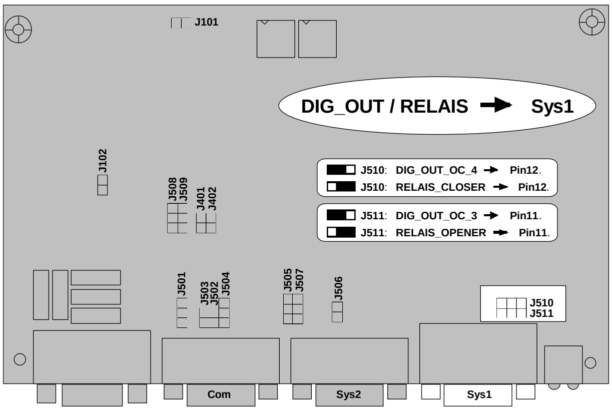

J101
DIG_OUT / RELAIS → Sys1
J510: DIG_OUT_OC_4 → Pin12.
J510: RELAIS_CLOSER → Pin12.
J511: DIG_OUT_OC_3 → Pin11.
J511: RELAIS_OPENER → Pin11.
J102
J508
J509
J401
J402
J501
J503
J502
J504
J505
J507
J506
J510
J511
Com
Sys2
Sys1

Figure 4-18: Configuration SYS1

Hint: If the relay output is needed, it is preferable to use the closer contact, because when using the opener contact the digital output signal for the position monitoring is no longer available.

By default the electronic unit is configure for operation with the output signal for the position monitoring (open collector output) and the relay closer contact.

SYS2 interface

The SYS2 interface is configurable for the operation with CAN bus. The CAN bus is not supported in normal operation and requires the implementation of a customer specific protocol. For further questions regarding CAN bus coupling please contact Sulzer Electronics AG.

The jumper positions for SYS2 must not be changed, otherwise the inputs for the set values will no longer be available.

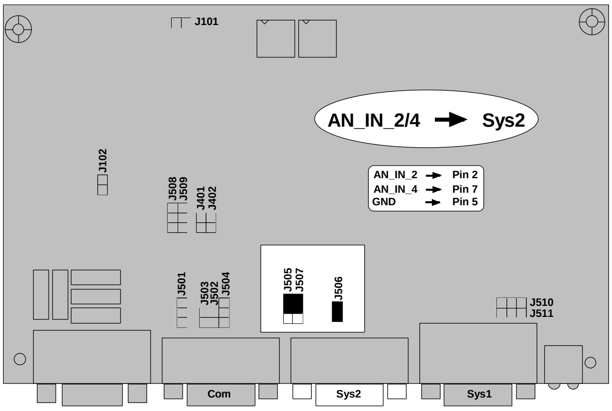


Figure 4-19: Configuration SYS2

By default the electronic unit is configured for operation with the analog inputs for the set values and the trigger signals. The jumper positions for SYS2 must not be changed, otherwise the inputs for the set values will no longer be available.

COM interface

The Com interface for configuring the electronic unit via the PC software *LinMotâ Talk* is normally operated in the RS232 mode. Upon request an RS485 interface is available. For further information regarding the RS485 interface please contact Sulzer Electronics.

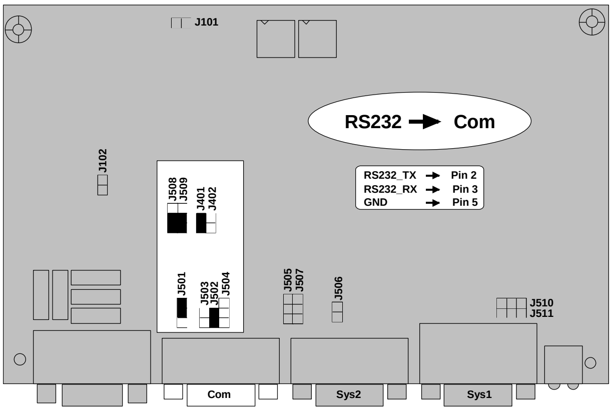


Figure 4-20: Configuration COM

By default the unit comes configured for operation with the RS232 interface.

4.9 Mechanical installation

The fixing of the electronic units is done by means of two M5 screws. The electronic unit can be screwed on to any backplane. The special fixing holes allow for easy mounting and removing of the unit.

The electronic units *LinMot® E400-AT* and *LinMot® E200-AT* are fitted with a heatsink and should be mounted in vertical position to allow better cooling. When mounting the units it is important to take into consideration the fact that the housing temperature can reach 60° (heatsink up to 70°). Sufficient room for heat dissipation should be provided. The power dissipation of the units can vary considerably depending on the application and should be determined for each case. An internal protection mechanism shuts the unit down when over temperature of the heatsink is detected.

4.10 Constructive notes on the installation of the linear motors

The linear drives of the *LinMot*® *P* family feature a slide bearing between the moving slider and the fix stator. The requirements on this slide bearing construction are exceptionally high, due to the enormous dynamic properties and acceleration of the drives. For these reasons the following points must be observed:

Transversal forces

Due to the surface pressure, a transversal forces have a stressing effect on the slide bearing and on the lifetime of the drive. For this reason care should be taken in the construction in order to reduce such transversal forces to a minimum, or to absorb them by means of an additional guide.

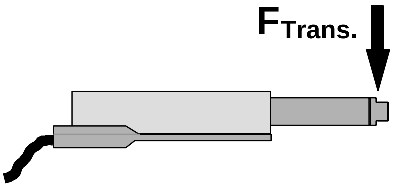


Figure 4-21: Transversal loads

The linear drives of the *LinMot*® *P* series are intended exclusively for operation as drive elements and are not allowed to be used as guide or bearing elements.

Parallelism error

In all constructions where the slider of the *LinMot*® *P* drives is used to move another longitudinally sliding machine part over direct coupling, an over determinate system arises.

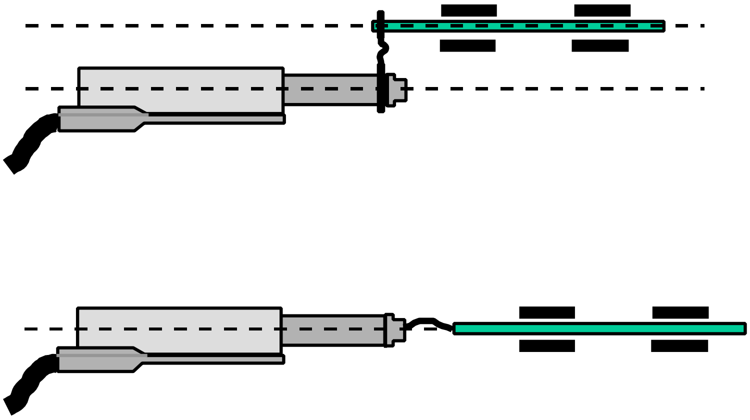


Figure 4-22: Compensation of parallelism errors

In order to compensate the inevitable offset and parallelism errors, a compensation capable coupling (represented here by a bent line) must be used between the slider and the moving machine part.

Horizontal mounting

If the linear motors are mounted horizontal, the notch of the stator should be on the lower side (see figures above). If mounted like this, the sliders mass is compensated by magnetical forces in the stator.

Mounting of the load

The load mass must be mounted in such a way, that only the end piece of the slider is held with the appropriate wrench (Caution: magnetical attraction). By no means should the sensitive slider tube or the slider end piece be clamped or used as a tightening tool.

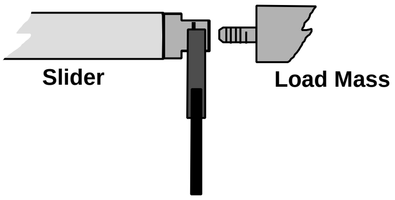


Figure 4-23: Load mass mounting

Handling the sliders

The sliders of the *LinMot*® *P* motors must be handled with great care. Even minor damage to the slider surface can cause a drastical reduction its lifetime. The slider is a high precision machine element consisting of a thin steel tube and Neodym magnets. As it is designed for loads in the longitudinal direction, even just hitting an iron plate due to the magnetical attraction can cause permanent damage to it.

Caution, the following points have to be observed when handling the slider:

- Under no circumstances may a damaged slider be used further, as this can lead to permanent and non reparable damage to the stator!
- The slider of the *LinMot*® *P* motors contains Neodym magnets, that can cause damage to magnetical data medium or sensitive electronic devices by merely coming close to them.
- When manipulating the sliders, hitting them against iron parts, tools, etc. must be absolutely avoided, as this can lead to permanent damage of the slider (surface damage, bending). Further, hitting against other ferrous objects represents a danger of injury (Bruised fingers, etc.).
- The slider of the *LinMot*® *P* motors can reach temperature values that can cause burns if touched.
- The sliders of the *LinMot*® *P* motors are fast moving machine parts. The user is responsible for taking all the measures necessary to avoid any contact and the relative danger of injuries to living beings (cover, protection from contact etc.).
- Accumulations of dirt, in particular of ferrous chips (magnetic attraction!) or dry running of the slider can considerably shorten the slider's lifetime.
- The sliders may only be stored or transported in the special plastic case (with cardboard inlay) or already inserted and blocked in the stator.
- The sliders must be protected from dirt (particularly metal chips, etc.).
- Maximal storage temperature: 70 °C
- Both slider and stator must be cleaned and subsequently re-greased if they become dirty (see chapter 7).

5. Configuration Software

The *LinMot*® Talk software is a MS-Windows based software designed to support the user in commissioning and configuring the *LinMot*® electronic unit. The software features a powerful, modular graphical interface that covers all the tasks involved with the operation of the *LinMot*® electronic unit. Great care has been taken during the development of the interface in order to make it easy to upgrade and comfortable to operate. The five main modules of the software are described shortly in the following text. A detailed description is found in the following chapters.

‘Control Panel’ With the ‘Control Panel’ the user can start and stop the software on the electronic unit. The electronic unit can also be reseted from here, and new software can be downloaded.

‘Parameter Inspector’ With the ‘Parameter Inspector’ the parameters of the *LinMot*® electronic unit can be changed easily. Complete parameter sets can be saved, loaded and printed.

‘Curve Inspector’ The ‘Curve Inspector’ is used to generate and manage reference curves for the *LinMot*® Electronic. Existing curves can be loaded, saved, merged and printed. New curves can be created in a simple way.

‘Oscilloscope’ The ‘Oscilloscope’ helps the user commissioning the *LinMot*® system. It allows real-time recording, subsequent on-screen viewing and printing of internal variables, like for example position set value and actual value.

‘Error Inspector’ With the ‘Error Inspector’ the user has the possibility to display the error log as well as the actual pending warnings and errors of the *LinMot*® electronic unit.

5.1 Installation requirements

The requirements for the installation of the *LinMot*® Talk software are summarized in the following table:

Resource	Minimal requirement	Recommended
computer	A personal computer with 80386 processor	personal computer with 80486 or Pentium processor
free memory	RAM: 8MB Harddisk: 4MB	RAM: 16MB Harddisk: 8MB
graphic card and monitor	VGA graphic card and 15” monitor	SVGA graphic card and 17” monitor
operating system	Windows 3.1 Windows 3.11 Windows 95 Windows NT 3.51 Windows NT 4.0	
pointing device	Windows compatible mouse	
free interfaces	serial RS232 interface	
disc drives	3½” disc drive	

Table 5-1: Installation requirements

5.2 Installation of the configuration software

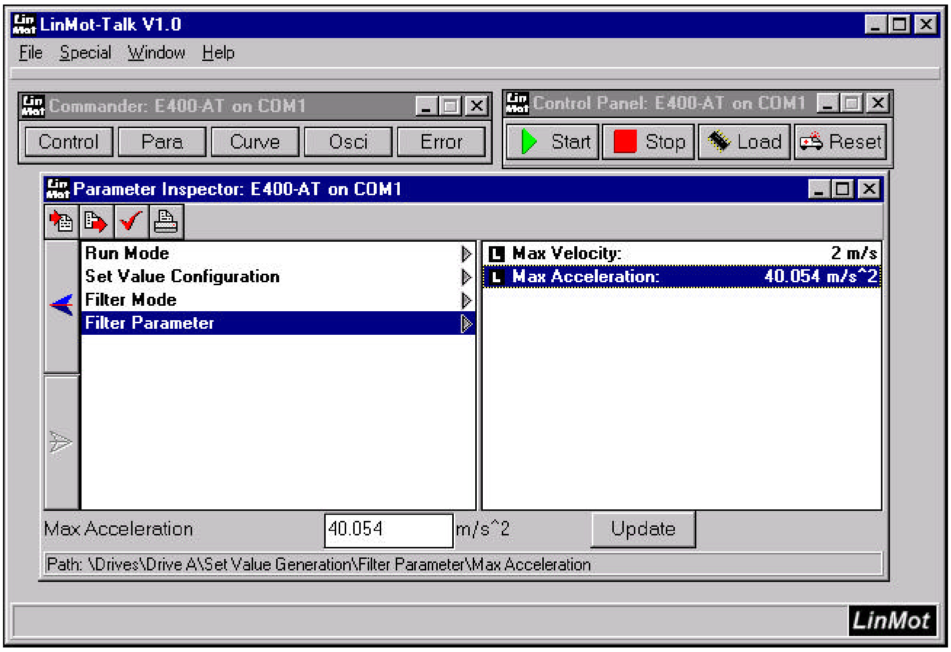
This chapter describes the installation of the *LinMot*® Talk configuration software.

Proceeding

1. Start the Windows operating system and insert the diskette labeled “*LinMot*® Talk/ Disk 1/1” in the corresponding drive.
1. Run “Setup.exe” from the inserted disk.
1. Follow the instructions of the installation program.
1. After completing the installation a new program group appears, with the name “*LinMot*® Talk”. The “README.TXT” file contains late breaking notes on the respective version.
1. Open the file “LINTALK.INI” with a text editor.
1. Remove the semicolons before the serial interface that you plan to use for *LinMot*® Talk. You can also have both interfaces activated. The following example shows a configuration file in which the COM1 interface is activated.

File “LINTALK.INI”:

```
[Devices]
Device01 = COM1
;Device02 = COM2
;Device03 = COM3
;Device04 = COM4
;Device05 = COM5
;Device06 = COM6
;Device07 = COM7
;Device08 = COM8
```


LinMot-Talk V1.0
File
Special
Window
Help
Commander: E400-AT on COM1
Control
Para
Curve
Osci
Error
Control Panel: E400-AT on COM1
Start
Stop
Load
Reset
Parameter Inspector: E400-AT on COM1
Update
LinMot

Run Mode
Set Value Configuration
Filter Mode
Filter Parameter

Max Velocity: 2 m/s
Max Acceleration: 40.054 m/s^2

Max Acceleration 40.054 m/s^2

Path: \Drives\Drive A\Set Value Generation\Filter Parameter\Max Acceleration

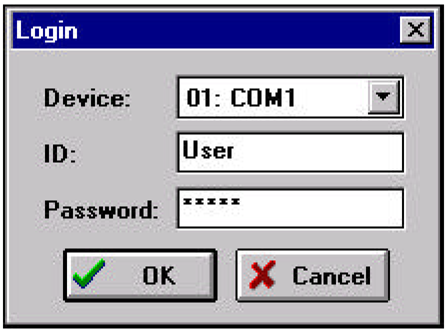
Figure 5-1: The *LinMot*® *Talk* software

5.3 Introduction

This and the following chapters describe the *LinMot*® Talk software and all its modules in detail. For a quick commissioning of a single motor refer to the step-by-step guide in Tutorial A.

5.4 Login

After starting *LinMot*® Talk, a blank window with a menu bar appears on the screen. As a first step the entitled user must log in the electronic unit. This is done with the "Login..." command under the "File" menu. After selecting the command a dialog box appears, as shown in figure Figure 5-2.



Login

Device: 01: COM1

ID: User

Password: *****

Figure 5-2: Login window

The interface to which the *LinMot*® electronic unit is connected can be selected here. Note that only the interfaces that were activated in the "Lintalk.ini" file will be displayed! The "ID" field should contain the name "User". The correct password must be entered at this point (see also the relative notes at the end of this section), after which it is possible to log in by clicking on the "OK" button. If the password was entered correctly the Commander window will appear after a few seconds. The Commander gives access to all the functions of the *LinMot*® Talk software.

Passwords

All *LinMot*® electronic units are delivered with no password set.

The password protects your electronic unit from illegitimate changes to the configuration. If you set a password, you should make a notice of it! A forgotten password can cause a good deal of annoyance at the wrong moment!

5.5 Commander

The Commander is the central control panel of the *LinMot*® Talk software. The single modules for the configuration and commissioning of the *LinMot*® electronic unit can be called from within it.

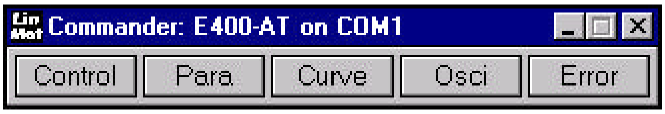


Figure 5-3: Commander window

5.6 Control Panel

With the control panel the user can start or stop the software on the *LinMot*® electronic unit. In case of a fatal system error there is also the possibility to reset the unit. When updating the basis software, the new software can be downloaded to the electronic unit with the “Load” command.

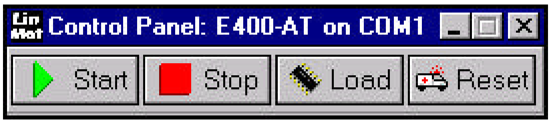


Figure 5-4: Control Panel window

“Start” Starts the operating software of the *LinMot*® electronic unit, if it is not already running.

“Stop” Stops the operating software of the *LinMot*® electronic unit. All motors are turned off.

“Reset” Executes a hardware reset on the *LinMot*® electronic unit. This is only necessary if a fatal system error has occurred. A fatal error is signalized on the front panel of the *LinMot*® electronic unit by the blinking of the “Fault” LED, and should never occur during normal operation.

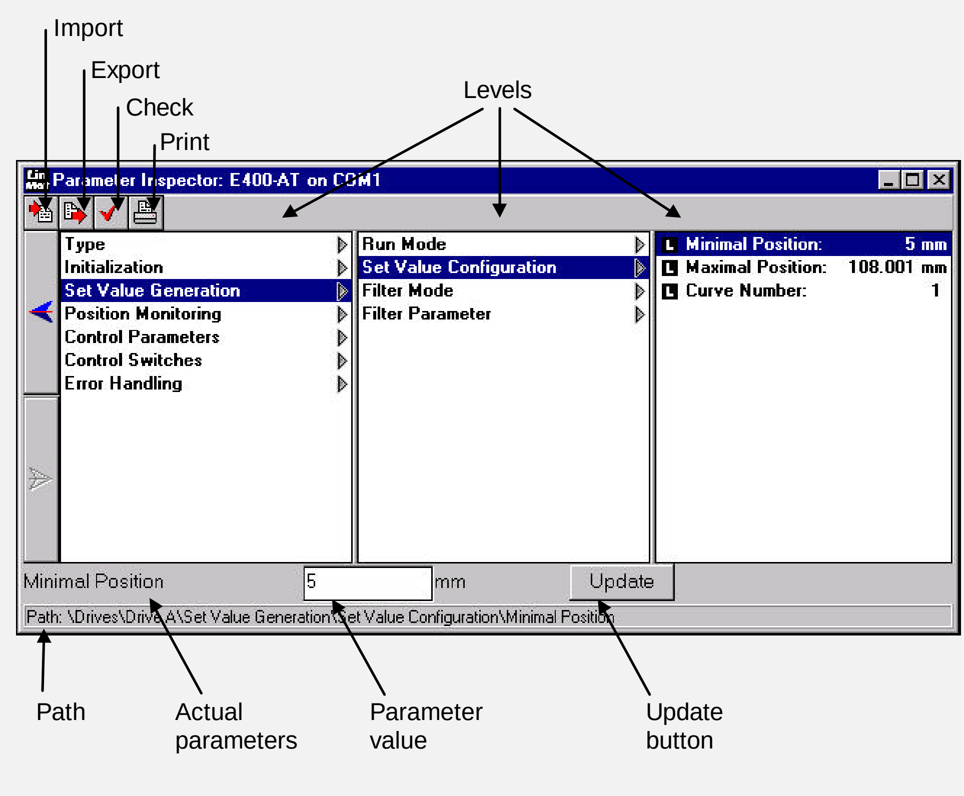
“Load” Loads a new basis software in the *LinMot*® electronic unit.

5.7 Parameter Inspector

The parameters of the *LinMot*® electronic unit can be displayed and edited with the Parameter Inspector:

- hierarchical display of the *LinMot*® electronic unit’s parameters
- Online and Offline modes
- Import and export of parameter configurations
- Copy and Paste of parameters and parameter sets
- Printing of parameter sets

Figure 5-5 shows an operating window of the Parameter Inspector. The window can be enlarged to make more levels visible. The user has also the possibility to navigate the parameter tree with the cursor keys.



Minimal Position: 5 mm
Maximal Position: 108.001 mm
Curve Number: 1

Minimal Position 5 mm

Path: \Drives\Drive A\Set Value Generation\Set Value Configuration\Minimal Position

Figure 5-5: Parameter Inspector window

5.7.1 Editing the parameters

The *LinMot*® software supports different parameter types, that the Parameter Inspector displays in different ways. There are four main groups.

Directories
▷

Directories contain collections of objects, just like in the Windows interface. These are either parameters or other directories. A directory can be recognized from the little arrow on the side of its name. It can be opened by clicking once on its name with the mouse.

Not all directories are displayed at all times. If, for example, “No Drive” is selected as the “Drive Type” for drive A it makes little sense to be able to edit the controller parameters for that drive. For this reason those parameters will not be displayed.

Value parameters

These parameters contain numerical values or character chains. When such a parameter is selected, its value appears in the lower status bar of the window. At this point a new value can be entered in the entry field. After clicking on the “Enter” button the value is tested to make sure it lies within the boundaries that were set. If the parameter is correct it is transmitted to the electronic unit and at the same time converted in the internal units. This can lead to small discrepancies between the value that was entered and the value that is actually displayed once it has been transmitted. The values entered are rounded to the values that are stored in the electronic unit (according to the resolution of the unit).

Live parameters
L

Live parameters are actualized immediately on the electronic unit after a change, whereas normal parameters are saved to the unit only after a new start.
For example, the controller parameters are live parameters. As soon as the controller settings are modified, the connected motor will behave accordingly. This way it is possible to change the controller parameters without having to stop and restart the software every time.
Live parameters are marked with a small L symbol on the left side of their name.

Selection parameters
○ / ◉

With selection parameters only one entry out of many can be selected. This is done by double-clicking or by hitting the “Enter” key. Selection parameters are marked with a small circle on the left side of their name. The selected parameter appears with a filled circle.

On/Off parameters
□ / ☒

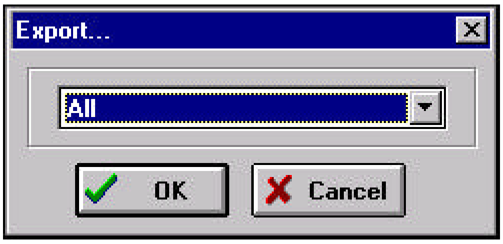
The On/Off parameters, as the names tells, can be turned either on or off by double-clicking. In most cases a whole group of these parameters is included in a directory. On/Off parameters are marked with a small square on the left side of their name, that is checked if the parameter is active.

5.7.2 Saving and loading parameter sets

The “import” and “Export” buttons allow loading and saving whole parameter sets.

Export

When exporting, it can be chosen whether all the parameters stored in the electronic unit or only the ones for a single motor should be saved to the PC.



All

Figure 5-6: Export dialog box

Import

When importing, a dialog box for the selection of a parameters file is displayed. In case of a motor parameter set it is further possible to choose which motor the parameters should be downloaded to.

File names for motor parameter sets end with the extension “*.dri”, while those for parameter sets for the electronic unit end with the extension “*.cfg”. The corresponding file extension must be selected when importing parameter sets. When exporting a set, the selection is done automatically.

5.7.3 Copy and Paste

In order to simplify the configuration of several motors, the software supports the Copy (Ctrl-C) and Paste (Ctrl-V) commands. The copy command copies the parameter actually selected. If this is a directory, then the entire directory is copied. When pasting, the destination parameter must be compatible with the copied parameter. If, for example, the directory “Drive A” was to be copied and pasted to the directory “Drive B”, the copy button (or Ctrl-C) would have to be clicked after selecting the directory “Drive A”. At this point “Drive B” can be selected and the key combination Ctrl-V would paste the copied parameter set from motor A to motor B.

5.7.4 Verifying parameter settings

The visibility of some parameters can depend on the settings of other parameters. For example, the initialization mode “Trig Turn Left” would only be visible if a stepper motor had been chosen as the “Drive Type”. When the user changes motor type it cannot be guaranteed that the correct initialization mode will be automatically selected.

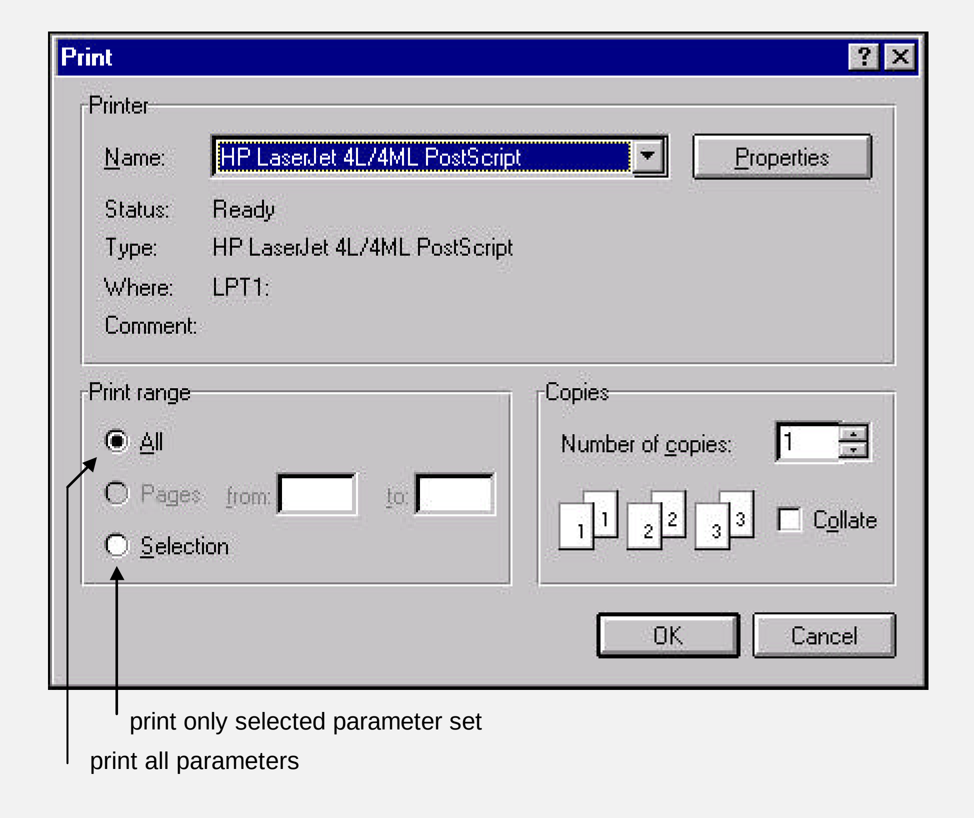
Check Configuration

✓

In order to quickly discover errors in the parameter configuration, a program function that checks the parameter configuration automatically is integrated. This function is called by clicking on the “Check Configuration” button. An incorrect or missing parameter setting is marked and displayed so that it can be corrected by the user. Subsequent clicking on the “check Configuration” button causes the display of the next incorrect parameter, or a message informing that the parameter configuration is consistent.

5.7.5 Printing parameter sets

The print process is initiated by clicking the “Print” button. If desired, only a single parameter set can be printed by selecting it.



Print

Printer

Name: HP LaserJet 4L/4ML PostScript

Status: Ready

Type: HP LaserJet 4L/4ML PostScript

Where: LPT1:

Comment:

Print range

● All

○ Pages from: to:

○ Selection

Copies

Number of copies: 1

□ Collate

print only selected parameter set

print all parameters

Figure 5-7: Printing parameters

5.7.6 Write-protected parameters

Some parameters cannot be configured by the user. This mostly concerns those parameters that affect the safety of the electronic unit or of the connected motors. For example, the monitoring of the supply voltage cannot be disabled. All parameters that cannot be edited by the user are displayed shaded.

5.7.7 Online mode

The user normally works in the Online mode. This means that the *LinMot*® electronic unit is connected directly to the PC via a serial interface. Changes in the parameters are transmitted to the *LinMot*® electronic unit over the interface. The Online mode is accessed by logging in and activating the Parameter Inspector from the Commander.

5.7.8 Offline mode

The Offline mode allows the editing of a parameter configuration that was saved beforehand. The electronic unit doesn't need to be connected to the serial interface for this operation. Changes to the parameters are not transmitted directly to the electronic unit, but the modified configuration can be saved. The Offline mode can be accessed by selecting the " Offline Edit..." command under the "File" menu.

Not all parameters are available in the Offline mode. The parameter "Time", for example, is not available, as it would normally display the time of operation of an electronic unit and there is no connection to an electronic unit in this mode. All the parameters that cannot be read in the Offline mode are marked with "n.a." (not available).

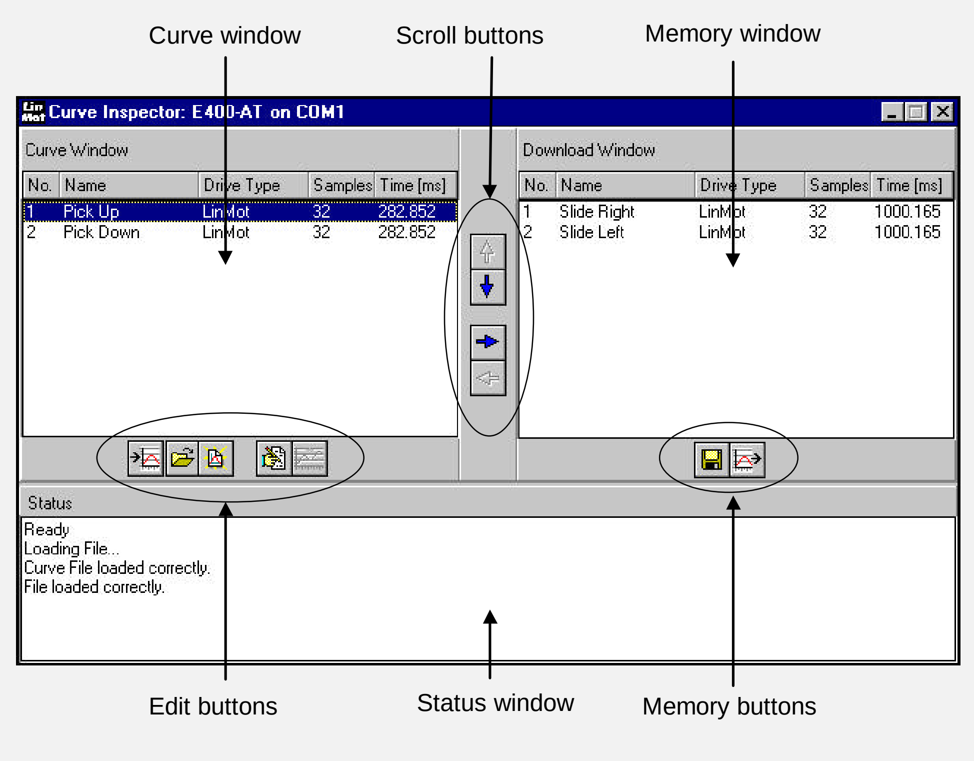
5.8 Curve Inspector

With the Curve Inspector the user disposes of a simple way to generate and modify reference curves for the different drive types. Curves can be loaded from files or from the electronic unit, edited and combined, and finally saved to a file or to the electronic unit.

This chapter is divided in two parts: the actual Curve Inspector and the Curve Editor.

5.8.1 User interface of the Curve Inspector

The interface of the Curve Inspector appears after clicking on the “Curve” button in the Commander window. Figure 5-8 shows a typical Curve Inspector window.



Curve Window

No.	Name	Drive Type	Samples	Time [ms]
1	Pick Up	LinMot	32	282.852
2	Pick Down	LinMot	32	282.852

Download Window

No.	Name	Drive Type	Samples	Time [ms]
1	Slide Right	LinMot	32	1000.165
2	Slide Left	LinMot	32	1000.165

Status

```
Ready
Loading File...
Curve File loaded correctly.
File loaded correctly.
```

Figure 5-8: User interface of the Curve Inspector

The single operational elements are described in the following chapter.

5.8.2 Operational elements of the Curve Inspector

Curve window

All curves that are loaded or newly generated are listed in this window. New entries are appended at the end of the list. The name, motor type (Drive Type), number of points (Samples) and time (Time) in which the curve is run are displayed. The number is only valid in this window and is used to distinguish missing or equal curves.

Memory window

This window displays the curves in the same order as they are selected and written to the electronic unit. The information displayed is the same as in the curve window.

Status window

The current actions and error messages are displayed in this window.

Edit buttons

Upload: A click on this button causes an upload of the curves that are stored in the connected electronic unit to the PC. Deppending on the number of curves and values stored in the electronic unit, the upload takes up to several seconds.

Open: Curve files can be loaded by clicking on the "Open" button. The file format (described in Tutorial D) is Excel-*.csv. During the load of the curves a log file is generated (*.log). This can be useful for tracing down format errors.

New: A new curve is generated and the Curve Editor is started automatically.

Edit: A click on the “Edit” button opens the Curve Editor which allows editing the selected curve. This button is only available when a single curve is selected in the curve window.

Join: All selected curves in the curve window are lined up and merged to a new, single curve. The velocity setting of the single curve sections is not modified. A warning is generated in case the end point of one curve does not correspond with the starting point of the next curve. In ceases where not all the curves are specified for the same drive type a warning is also generated.

When merging curves, some points must be interpolated. This could lead to minor changes in the curves. This effect can be minimised by selecting a higher number of points for the curves.

Memory buttons

Save: All the curves in the ”Save Window” are written to a file in the same order as displayed. The file can be stored either in the *.csv or in the *.h86 file format. It is recommended to always save the curves in the *.csv format, as only this format stores all the information relative to the curve in the file. Curves saved in this format can also be edited in Excel. The *.h86 format is used to save the curve in such a way that it can be written to the electronic unit with the “Load” command from the “Control Panel”.

Download: All the curves in the memory window are written to the electronic unit in the same order as displayed. First, the electronic unit is checked for memory availability for the selected curves. After this, the unit is stopped and the data is written to it. When writing new curve data to the electronic unit the old curves are overwritten.

Scroll buttons

↑ **Up**: All the selected curves are shifted up by one position by clicking on this button.

↓ **Down**: All the selected curves are shifted down by one position by clicking on this button.

→ **Right**: All the curves selected in the curve window are moved to the memory window and appended to the list in the same order.

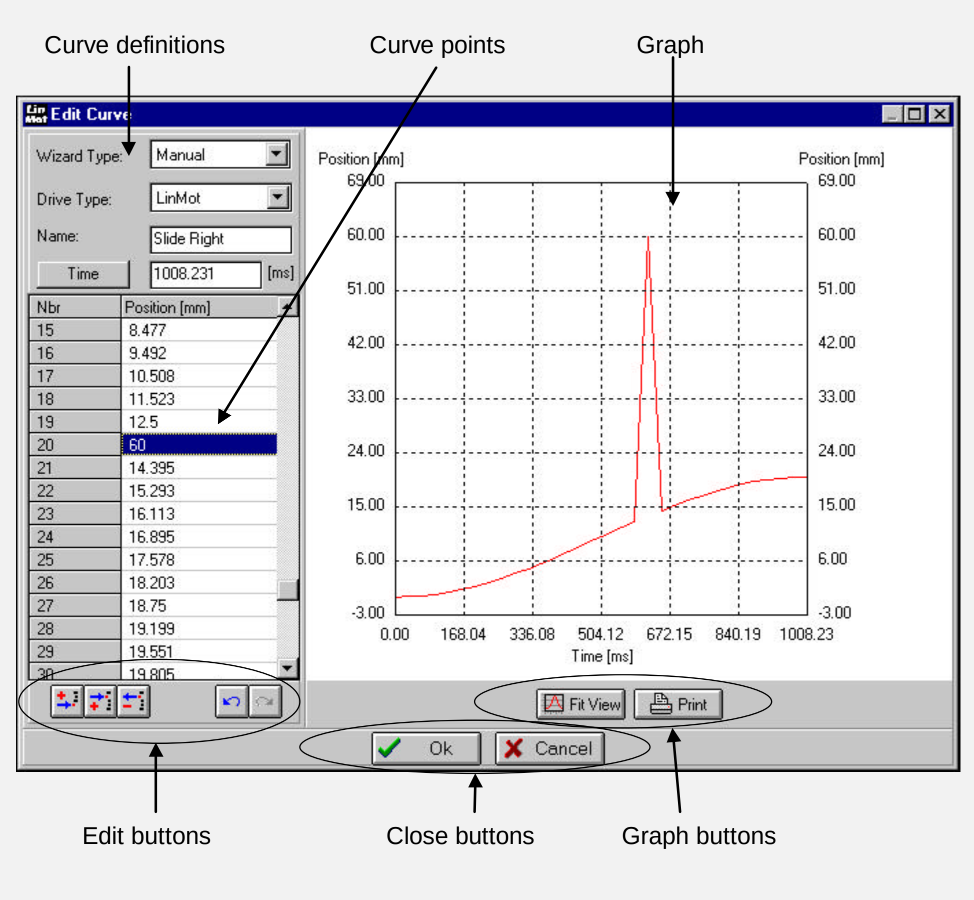
← **Left**: All the curves selected in the memory window are moved to the curve window and appended to the list in the same order.

Keyboard commands

Delete: The selected curves are deleted. Each modified curve can be saved separately by the user.
Ctrl + C: Copies the selected curves in a buffer.
Ctrl + V: Pastes the curves in the buffer to the selected curve.
Shift + Insert: Same function as Ctrl-V.

5.8.3 Curve Editor interface

Reference curves can be created in different ways with the help of the “Wizards”. The input possibilities and the interface are adapted accordingly. The Curve Editor is used to edit the curves. Depending on the type of curve it can have one of the following interfaces.



Wizard Type: Manual

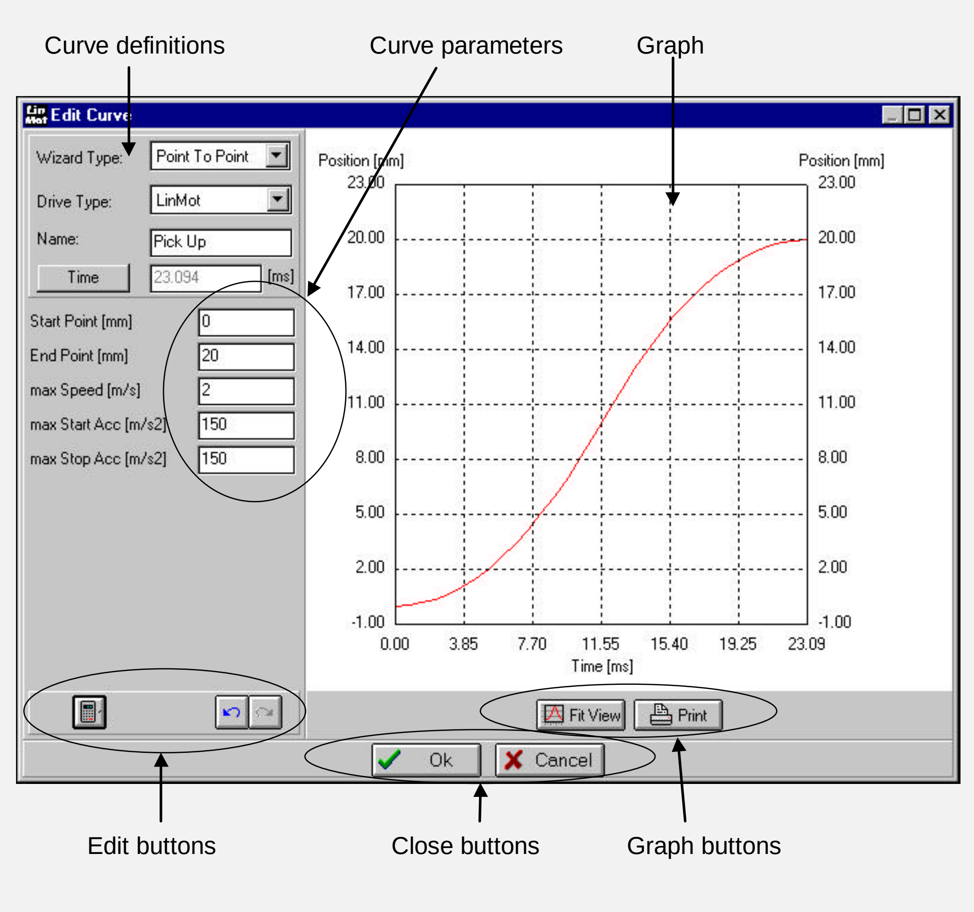
Drive Type: LinMot

Name: Slide Right

Time 1008.231 [ms]

Nbr	Position [mm]
15	8.477
16	9.492
17	10.508
18	11.523
19	12.5
20	60
21	14.395
22	15.293
23	16.113
24	16.895
25	17.578
26	18.203
27	18.75
28	19.199
29	19.551
30	19.805

Figure 5-9: Manual creation of curves



Wizard Type: Point To Point

Drive Type: LinMot

Name: Pick Up

Time 23.094 [ms]

Start Point [mm] 0

End Point [mm] 20

max Speed [m/s] 2

max Start Acc [m/s2] 150

max Stop Acc [m/s2] 150

Figure 5-10: Curve Editor for calculated curves

The settings are edited in the left panel of the window, while the curve is displayed graphically in the right panel.

5.8.4 Operational elements of the Curve Editors

The operational elements of the Curve Editor and their functions are explained in this chapter. The elements that are always visible are described first.

Graph

The curve is displayed graphically in this window. Changes are displayed immediately for manually edited curves (“Manual” wizard) or after a curve calculation for calculated curves. The curve can be selected and moved with the mouse.

Graph buttons

Fit View: The axis of the graph are scaled automatically to fit the whole curve in the display.

Print: The curve graph is printed with the currently selected setting and scaling.

Close buttons

✓ Ok

OK: This closes the Curve Editor and the curve is taken with the actual settings. If the “OK” button is not available, the curve must first be calculated by clicking on the “Calculate” button.

✗ Cancel

Cancel: This closes the curve editor discarding any changes made. A confirmation is requested in case the curve was modified.

Curve definitions

Wizard Type: With the “Wizard Type” the user can select in which way he wants to create the reference curve. The possibilities are: “Manual”, “Ramp”, “Sine” and “Point to Point”. Chapter 5.8.5 describes the wizards in detail.

Drive Type: This specifies for which type of drive the curve is to be generated. The following choices are available: “LinMot Pxx-xx", "Stepper" and "Magnet".

Name: The name of the curve is free, as long as it doesn’t exceed 22 characters in length.

Time: The time unit display in the neighboring editable field can be selected with the “Time” button between period [ms] and frequency [Hz]. The speed at which the curve will be executed is set in this field. In the “point to Point” wizard the user cannot specify the time, as it is calculated.

If the drive type selected in the creation of the curve does not correspond with the actual drive type, the electronic unit will generate an run-time error (“Curve Error”) when the curve is called.

Edit buttons

Add: The number of points that should be added at the end of the curve can be entered in the dialog box that is displayed by clicking on this button. The inserted values are set automatically to the value of the last edited point. This button is only visible for the “Manual” wizard.

Insert: The number of points that should be inserted at the selected point in the curve can be entered in the dialog box that is displayed by clicking on this button. The inserted values are set automatically to the value of the last point. This button is only visible for the “Manual” wizard.

Remove: All the selected points are removed from the curve. This button is only visible for the “Manual” wizard.

As a result of rounding, changing the number of points can slightly affect the speed at which the curve is executed. This effect is more evident in the case of very slow curves, or curves with few points. The values displayed during editing are not the actual entered values, but the ones actually usable by the electronic unit.

Calculate: The curve is calculated according to the selected wizard and the settings made. A dialog box with a suggested number of points is displayed. The user can accept the value or set a specific number of points for the curve. This button is visible for the “Ramp”, “Sine” and “Point to Point” wizards.

Undo: The last entry is undone. A maximum of 50 undo steps can be executed.

Redo: This restores the last action that was undone.

Curve Point List

This table displays the single points of the curve in the “Manual” wizard. Points can be added, inserted and removed with the “Add”, “Insert” and “Remove” buttons. The number format and the value are checked after entering.

Curve parameters

Start Point: The starting point of the curve is defined here. The unit depends on the selected “Drive Type” and is adjusted automatically. "Start Point" is visible for the "Ramp", "Sine" and "Point To Point" wizards.
End Point: The ending point of the curve is defined here. The unit depends on the selected “Drive Type”. "End Point" is visible for the "Ramp", "Sine" and "Point To Point" wizards.
max Speed: The maximum speed that is allowed in the curve can be set with “max Speed”. The unit depends on the selected “Drive Type”. "max Speed" is only visible for the "Point To Point" wizard.
Max Start Acc: The maximal acceleration allowed when the curve is started is defined by "Max Start Acc". The unit depends on the selected “Drive Type”. “Max Start Acc” is only visible for the "Point To Point" wizard.
Max Stop Acc: The maximal acceleration allowed when the curve ends is defined by "Max Stop Acc". The unit depends on the selected “Drive Type”. “Max Stop Acc” is only visible for the "Point To Point" wizard.

5.8.5 Creating curves with the wizards

The wizards are used to enter reference curves in different ways. This gives the user the possibility to define the possible motion profiles and current profiles with little effort.

Manual

With this wizard every value in the curve to be created can be entered manually. The user can set the number of points in the curve and the time corresponding to one run of the curve. All changes are displayed automatically in the graph window during editing.

Ramp

This generates a curve for a linear movement from the “Start Point” to the “End Point”, to be executed in the time specified.

Sine

This generates a sine formed curve from the “Start Point” to the “End Point”, to be executed in the time specified.

Point To Point

This generates a curve for a jump from the “Start Point” to the “End Point”, for which the maximum start and stop accelerations are limited by “max Start Acc” and “max Stop Acc”. The maximum speed is given by “max Speed”. The time needed to execute the movement defined in the curve can only be read, as it is calculated by the wizard according to the limits set.

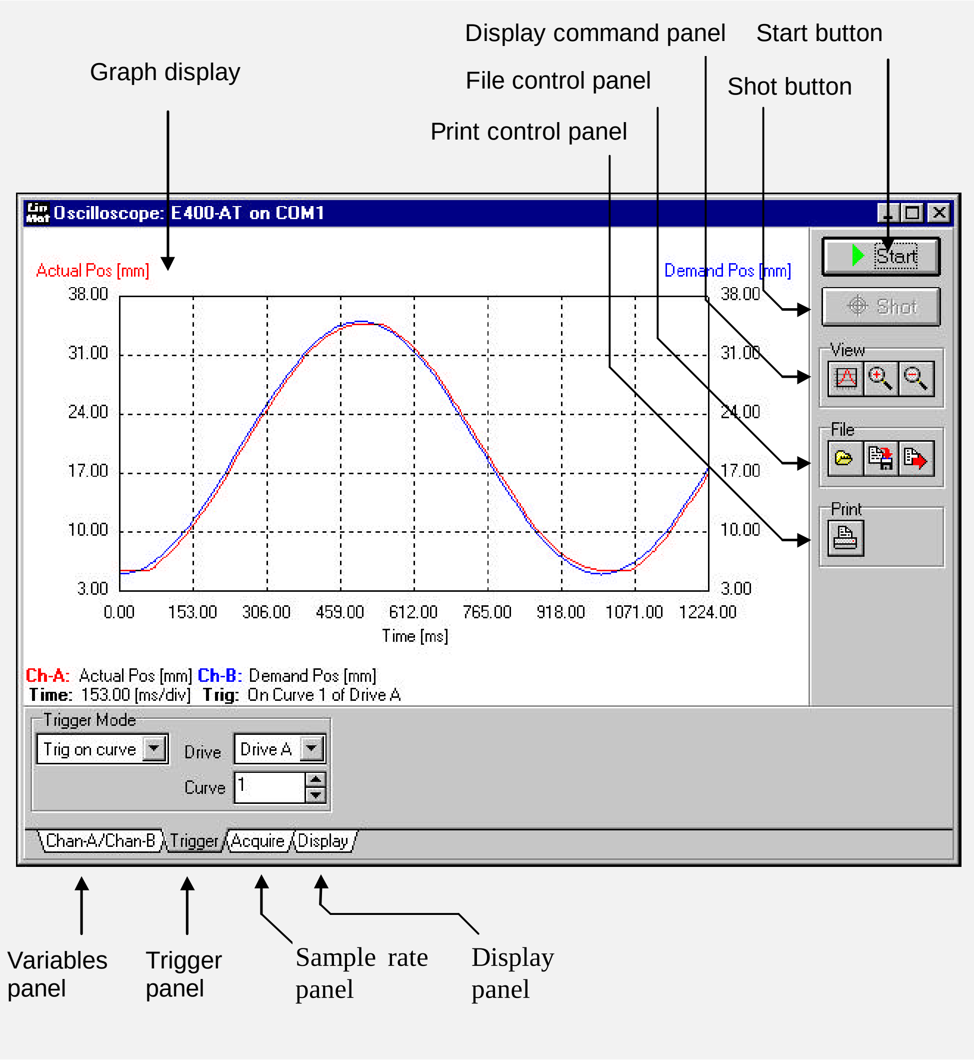
5.8.6 Defining reference curves in Excel

Users with high requirements in the creation of reference curves can create and/or edit curves with the spreadsheet calculation program Excel.
When editing curves in Excel it is important that the format of the curve files is maintained. For this reason this way is only recommended for experienced users.
Tutorial D shows an example of how curves can be edited with Excel.

5.9 Oscilloscope

The Oscilloscope is an important resource that helps the user in the commissioning of his drive system. The Oscilloscope offers the possibility to record data on the *LinMot*® electronic unit in real time and then transmit it to the PC to be displayed. It offers the following functions:

- two independent channels
- 256 data values for each channel
- variable sample rate from 400µs to 26.2s
- various trigger modes



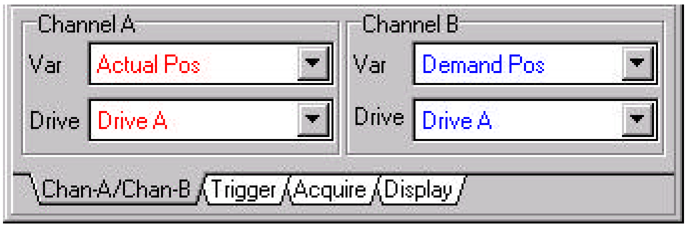
Ch-A: Actual Pos [mm] Ch-B: Demand Pos [mm]
Time: 153.00 [ms/div] Trig: On Curve 1 of Drive A

Trigger Mode: Trig on curve Drive: Drive A Curve: 1

Figure 5-11: Working with the Oscilloscope

5.9.1 Setting the recording variables

By setting the recording variables the user specifies which variables are to be sampled during the recording process. The variables panel must be activated before setting the variables (see also Figure 5-11). For each of the two channels the user can select the variable from a list. In case of drive variables the user must also specify the corresponding drive.



Channel A
Var: Actual Pos
Drive: Drive A

Channel B
Var: Demand Pos
Drive: Drive A

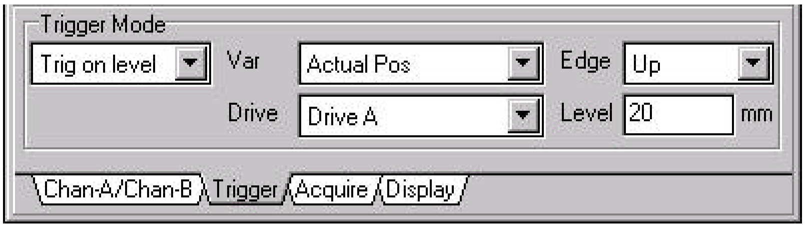
Figure 5-12: The variables panel

5.9.2 Setting the trigger mode

The trigger mode defines when the recording should be started. The following trigger modes are available:

Trig on Level

The recording is started when the selected variable exceeds, res. falls below the threshold specified. In Figure 5-13 the trigger is set to start the recording when the actual position of drive A exceeds the given position.

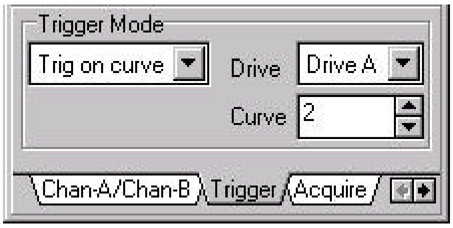


Trigger Mode: Trig on level
Var: Actual Pos
Drive: Drive A
Edge: Up
Level: 20 mm

Figure 5-13: Trigger mode 'Trig on level'

Trig on Curve

The recording is started when the selected curve is started. In Figure 5-14 the trigger is set to start the recording when curve 2 is started on drive A.

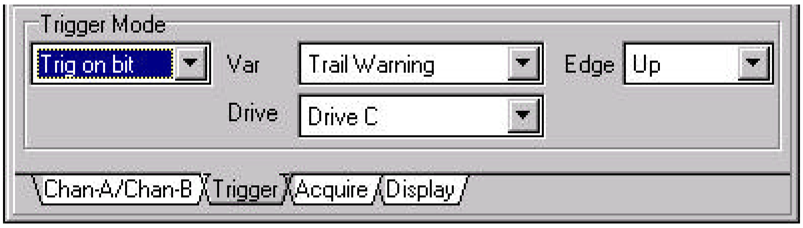


Trigger Mode: Trig on curve
Drive: Drive A
Curve: 2

Figure 5-14:Trigger mode 'Trig on curve'

Trig on Bit

The recording is started upon a state change of the selected bit. In Figure 5-15 the trigger is set to start the recording when a following error occurs on drive C. This setting is used mostly in combination with the "Pretrigger" mode (see also chapter 5.9.3).



Trigger Mode: Trig on bit; Var: Trail Warning; Edge: Up; Drive: Drive C

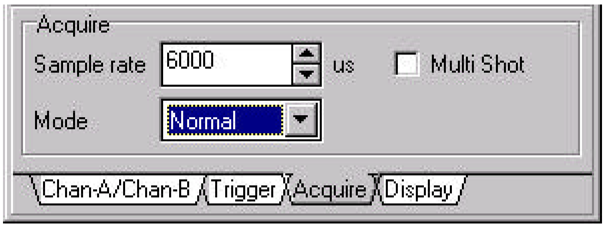
Figure 5-15: Trigger mode 'Trig on bit'

Trig Manual

The recording is started upon clicking on the Start button.

5.9.3 Determining the sample rate

The rate at which data values are sampled is specified in this field. This setting is always valid for both channels. It is further possible to select the sampling mode.



Acquire
Sample rate: 6000 us
□ Multi Shot
Mode: Normal

Figure 5-16: Setting the sample rate

Normal

In the "Normal" mode the selected channels are recorded without delay after the trigger signal. This means the recording is started as soon as the trigger condition is satisfied.

Pretriggered

It is often interesting for the user to be able to record events that happened just before the trigger signal was released. A typical example would be the occurrence of a following error. These events can be recorded in the pretrigger mode. The setting of a pretrigger is given as a percentage. A pretrigger of 80% would cause 80% of the recording time to be dedicated to events preceding the trigger condition and 20% of it to be dedicated to events following the trigger.

Delayed

In some situations it can be advantageous to start recording data after a certain delay following the trigger condition. In the "Delayed" mode the user can set a value for the delay time.

Multishot

It is sometimes possible that during the commissioning of a system an error occurs during one of maybe ten movements. In this case it would be difficult to detect and record the error with the Oscilloscope, in order to be able to analyze the signals and understand the cause of the error. The "Multishot" function gives the user the possibility to record this kind of errors. As in the other modes, a trigger condition must be set here, too. The difference is that the data are not transmitted to the PC automatically after the recording is done. The user initiates the transmission manually by means of the "Shot" button. This enables the user to wait for the error to happen and then display the recorded data on the PC by clicking on the "Shot" button. The "multishot" function is only available when neither the trigger mode "Manual" nor the recording mode "Pretrigger" are selected.

5.9.4 Adjusting the display

The way the recording data are displayed on the screen can be adjusted from the display panel. The adjustable parameters are:

- curve color
- curve offset (relative to the center of the graph display)
- magnification of the curve

The curves can be moved and zoomed in or out using the mouse or the buttons in the display panel.

Fit View

After the recorded the data have been loaded from the electronic unit up to the PC, the data are displayed in the display window. If the data does not fit in properly, the 'Fit view' function may be used for an automatic adjustment. The all data will be displayed.

5.9.5 Starting and stopping the Oscilloscope

Start

After clicking on the "Start" button, all the values are checked and sent to the *LinMot*® electronic unit. At this point the software on the *LinMot*® electronic unit waits until the trigger condition is satisfied and then starts the recording. As soon as the values have been recorded the *LinMot*® electronic unit notifies the PC that the data are ready and available. The PC then reads the data from the electronic unit and displays it. There are however three exceptions to this behavior.

Multishot

When the "Multishot" function is selected, the Start button must be clicked first, in order to start the Oscilloscope on the *LinMot*® electronic unit. When the desired event occurs the last event recorded can be transmitted to the PC by clicking on the "Shot" button. If the "Shot" button is clicked before the recording is finished, the recording is finished first.

Pretrigger

If the recording mode is set to "Pretrigger" and the trigger mode is "Manual", the Oscilloscope is started on the *LinMot*® electronic unit by clicking on the "Start" button. The trigger is then released by clicking on the "Shot" button. As soon as the recording is finished, the data are transmitted to the PC and displayed.

Abort

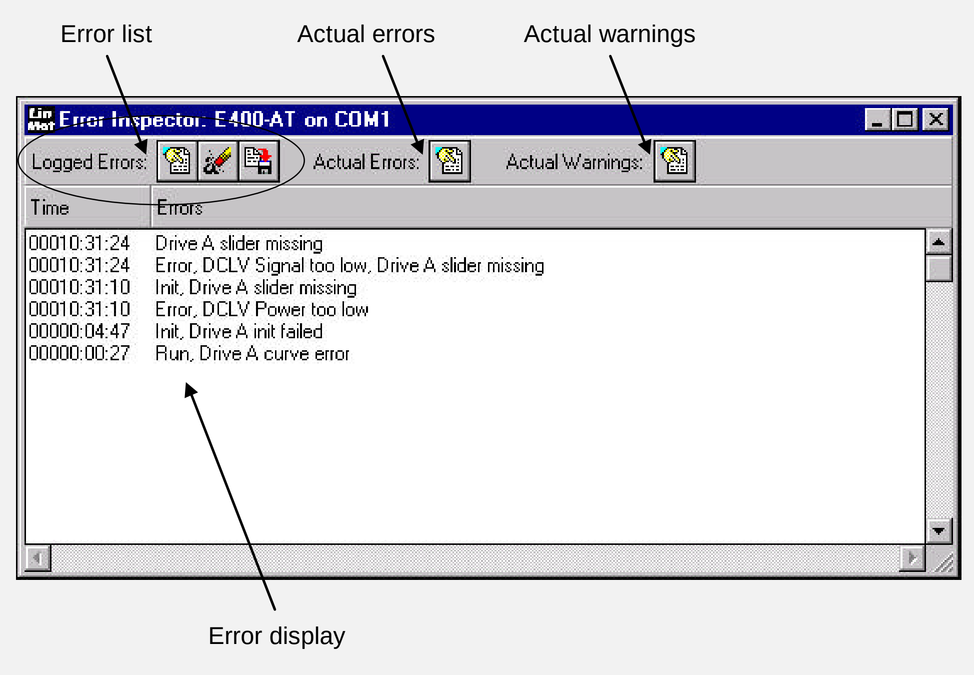
After the recording has been started by clicking on the "Start" button, the button text changes to "Abort". The "Abort" button can be used to abort the current recording.

5.9.6 Printing recorded curves

The actual graph can be printed to any printer by clicking on the "Print" button.

5.10 Error Inspector

With the Error Inspector the user of the *LinMot*® electronic unit can obtain information about warnings and errors occurred during operation. The Error Inspector offers the possibility to inspect the actual errors and pending warnings as well as the last 16 errors and warnings that were logged in the *LinMot*® electronic unit.
The following figure shows the Error Inspector window:



```
Time            Errors
00010:31:24     Drive A slider missing
00010:31:24     Error, DCLV Signal too low, Drive A slider missing
00010:31:10     Init, Drive A slider missing
00010:31:10     Error, DCLV Power too low
00000:04:47     Init, Drive A init failed
00000:00:27     Run, Drive A curve error
```

Figure 5-17: Interface of the Error Inspector

5.10.1 Operational elements of the Error Inspectors

The following sections describe the groups indicated in the Figure above:

Error display

The error messages contain information about system time, state of the basis software, drive type and type of error. The time is given in the "HHHH:MM:SS" format, where "HHHH" is the number of hours, "MM" the number of minutes and "SS" the number of seconds.

Error list

These buttons control the access to the errors stored in the electronic unit.

Show: The "Show" command reads the last 16 error entries from the *LinMot*® electronic unit and displays them in the error display. Note that not all the errors that lead to the "Error" state are saved and viceversa. The errors that should be logged are specified in the parameter tree under the System\Error Handling\Logging mask.

Clear: This command deletes the entire list from the *LinMot*® electronic unit. A confirmation is asked before executing this command.

Save: All the errors listed in the error display can be saved to a file with the extension "*.err". This button is only available right after having clicked on the "Show" button.

Actual errors

Show: Two sections are displayed after clicking on the "Show" button of "Actual Errors":
The section "ERROR THAT CAUSED TRANSITION TO ERROR STATE" shows which error caused the transition to the "Error" state.
The section "PENDING ERRORS" lists all the errors active at the moment of reading.

If the "Show" button is clicked when the basis software is not in the "Error" state, nothing appears in the display window.

Actual warnings

Show: Clicking on the "Show" button of "Actual Warnings" causes the actual warnings to be read and displayed.

The actual errors and warnings can only be displayed when the software on the *LinMot*® electronic unit is started.

5.11 Tutorial A: First Steps

This chapter shows an example of how to put the entire system into operation. The example shows how to configure a *LinMot*® *P* drive to execute a given curve endlessly after having done a reference run.

Requirements

- PC with *LinMot*® Talk software installed, as described in chapter XXX.
- RS232-cable for the connection of the PC with the *LinMot*® electronic unit.
- *LinMot*® electronic unit (E100/E400/E1000/E4000).
- Power supply for the *LinMot*® electronic unit.
- Supply cable

Cabling

The proper connection of a *LinMot*® electronic unit to a power supply is described in chapter 4. It is essential, that the electronic unit is grounded correctly before putting it into operation. If the electronic unit is grounded correctly, a linear motor can be connected to the “Mot A” port on the electronic unit. Then the electronic unit can be connected to the PC with a standard RS232 cable.

5.11.1 Commissioning example

In this example a *LinMot*® P drive is connected to the “Mot A” port of an electronic unit. It will be configured to follow reference curve endlessly.

Steps

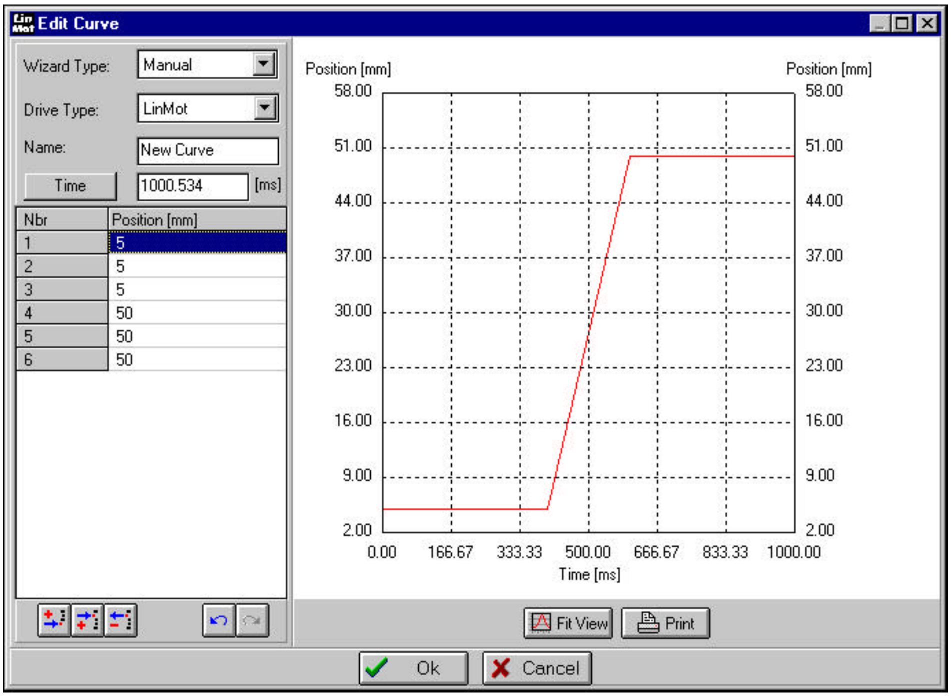
1. Interconnect the whole system as described in chapter 4.
1. Start the *LinMot*® Talk program on the PC.
1. Select “Login...” from the “File” menu. The login dialog box appears. By default no password is set in the electronic unit and the communication can be started by clicking on the “OK” button. After successful login, the Commander is displayed. The Commander gives access to all the modules of the *LinMot*® Talk program. In this example we will use the Parameter Inspector and the Curve Inspector modules.
1. By pushing the “Param” button, the Parameter Inspector is started. All available parameters are dispalyed hierarchically. The user can easily edit the parameters. To show more hierarchical levels, enlarge the window.
1. The following description assumes the standard parameter configuration. This configuration can be recovered by clicking on the “Import” button and then selecting the “\Lintalk\Defaults\E400AT.CFG” file. This is the default configuration that comes with the *LinMot*® electronic unit when delivered.
1. Change the parameter “\Drives\Drive A\Type” from “No drive” to “LinMot Pxx-23” or “LinMot Pxx-37”.
1. Change the parameter “Drives\Drive A\Set Value Generation\Run Mode” to “Continuous Curve”.
1. On the Commander panel push “Curve” to start the Curve Inspector. Pushing “Load Curves” allows to load an existing curve-file. For an easy and convenient commissioning an example is given, the file “CURVES.CSV” in the directory “LinTalk1\Examples”.
1. After loading the file, use the mouse to select the curve. Push the arrow-right button to move the curve to the “Download” window. Push “Download” to write the curve to the electronic unit.
1. On the “Control-Panel” push first “Stop” and then “Start”. The linear drive initializes and then executes the curve endlessly.

5.12 Tutorial B: Operating the Oscilloscope

In this section two real application examples are demonstrated and discussed. In order to have a well specified starting point it is necessary to create a reference curve, load it on the electronic unit and configure the unit accordingly.

5.12.1 Creating a curve

We want to create a simple curve that will be executed by the motor during the recordings with the Oscilloscope. It is assumed that the user is already familiar with the Curve Inspector and the Curve Editor.
We open the Curve Inspector and create a new curve. Here we set “Drive Type” to “LinMot”. Under “Wizard Type” we select “Manual” and we enter 1000 ms for “Time”. Then we enter six values, the first 3 set to 5 mm, the others set to 50 mm.



Wizard Type: Manual

Drive Type: LinMot

Name: New Curve

Time 1000.534 [ms]

Nbr	Position [mm]
1	5
2	5
3	5
4	50
5	50
6	50

Figure 5-18: The curve created

We now close the curve (with the “OK” button) and write it to the electronic unit (by moving it in the download window and then clicking “Download”).

5.12.2 Configuring the electronic unit

The easiest way is to start from the standard or default configuration. This can be imported in the Parameter Inspector by clicking on “Import”. It is important to load a configuration that is compatible to the electronic unit being used (under Defaults\). Now you can select the type of motor connected (LinMot Pxx-23 or LinMot Pxx-37) under Drives\Drive A\Type and then select the “Continuous” mode under Drives\Drive A\Set Value Generation\Run Mode. With this the necessary settings are made. At this point clicking on “Start” in the Control Panel initializes the motor and starts running the curve cyclically.

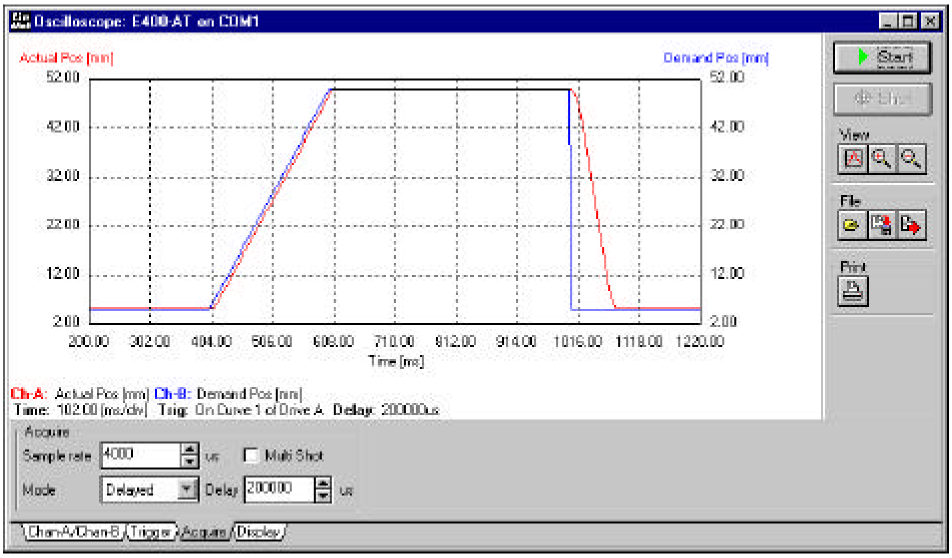
5.12.3 Oscilloscope configuration for the recording of the position set value

In this example the Oscilloscope will be used to display the actual value and the set value of the position. For this we open the Oscilloscope by clicking on the “Osci” button in the Commander. The variables for the two channels are found beneath the empty graph windows. “Actual Pos”, ”Drive A” should be selected for channel A, while “Demand Pos”, ”Drive A” should be selected for channel B.

We now switch to the entry panel for the trigger by clicking on “Trigger” in the lower part of the Oscilloscope window. As the motor executes the curve cyclically and a mark is always set at the beginning of a curve, the “Trig on Curve” offers an easy way to synchronize the start of the recording with the start of the curve. We select “Trig on Curve”. The entry fields for “Drive” and “Curve” appear. For “Drive” we should select “Drive A” and for “Curve” we should select “1”. Whit it we have defined that the recording will start as soon as motor A starts executing curve 1.

We now switch to the “Acquire” panel by clicking on “Acquire” in the lower part of the Oscilloscope window. Here we can select the sample rate and the sample mode. We set a sample rate of 4000 µs, so that we can record approximately a whole period. (the curve is executed in 1000 ms and the Oscilloscope records 256 values). We set the mode to “Delayed” and enter a delay of 200’000 µs. Multishot is not set. This way we start the recording 200 ms after the curve has started, in order to obtain a better resolution in the display of the rising edge, the most interesting part of the curve.
We don’t need to modify any settings under Display, unless you wish to change the plot color, the offset or the axis scaling.

We are now ready for a recording. We start the motor (“Start” in the Control Panel) and click on the “Start” button in the Oscilloscope when the motor is executing the curve. It takes only a few seconds before the values are loaded from the electronic unit. When no more text appears in the graph window, click on the “Fit View” button and you will see the following display:



Ch-A: Actual Pos [mm] Ch-B: Demand Pos [mm]
Time: 102.00 [ms/div] Trig: On Curve 1 of Drive A Delay: 200000us

Acquire
Sample rate 4000 us ☐ Multi Shot
Mode Delayed Delay 200000 us

Figure 5-19: Recording actual position and the set value of a cyclical curve

With the Zoom buttons under "Display" you can change curve sections and graph scaling. The "Fit View" button restores the display as shown in the previous section.

When "Multi Shot" is selected under "Acquire", a new recording is started as soon as the next trigger comes. This happens until the "Shot" button is clicked. "Shot" can be clicked manually when an error that should be analyzed occurs.

Two other modes are available under "Mode" in "Acquire": "Normal" and "Pretriggered". In the "Normal" mode the start delay is 0 s while in the "Pretriggered" mode values are been recorded already before the actual trigger. It is important to note, that in the "Pretriggered" mode enough time must have elapsed between the moment the "Start" button was clicked and the actual trigger signal. Only in this case all the values are valid. If this time is too short, the buffer could still contain data from a previous recording. The minimal time that should elapse, can be calculated from the sample rate, the pretrigger value and the information that 100% corresponds to 256 samples.

5.12.4 Oscilloscope configuration for the following error monitoring

In the first example we have seen how a recording can be triggered by a regular signal, such as the curve beginning. In this example we want to take a look at how to display exceptional events, such as warnings and errors.

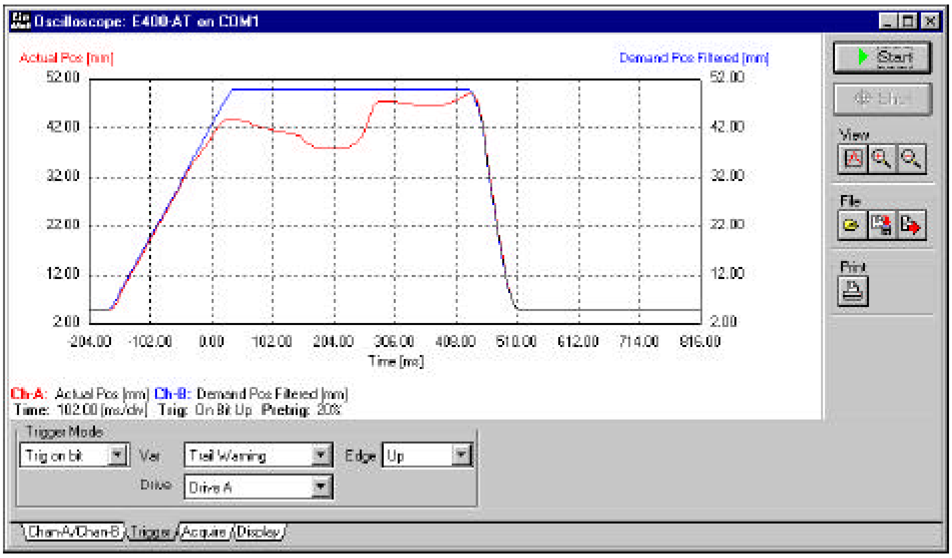
In order to reduce the changes to a minimum we will use the same motor with the same configuration. We want to record the movements of the motor in a case of following error.

On channel A we leave the setting "Actual Pos" to "Drive A", while on channel B we select "Demand Pos Filtered" on "Drive A". "Demand Pos" is the position set value that is read directly from the reference curve. As the set values are limited by the maximum allowed speed and acceleration settings, "Demand Pos Filtered" represents the values that are actually used by the motor controller. The speed and acceleration limiters can be enabled, disabled and configured in the parameter tree.

As a next step we choose the trigger source. We set the "Trigger Mode" to "Trig on bit" and select "Follow Warning", "Drive A" as "Var". We select "Up" for the active edge. This way the recording will start when a following error occurs on motor A.

We switch to the entry fields under "Acquire" and leave the sample rate at 4000 µs, but we set the mode to "Pretriggered" and "Pretrigger" at 20%. 20% of the recorded values will represent a recording of the signals before the trigger.

We now want to test how the Oscilloscope behaves with these settings. For this purpose we start the motor (by clicking "Start" in the Control Panel). When the motor executes the curve, we start the recording with the Oscilloscope. The message "Waiting for trigger" appears. If we hold the slider or force it to some position the recording will start. Again, it takes some time before the values are read. Click "Fit View" and the graphs obtained could appear as follows:



Ch-A: Actual Pos [mm] Ch-B: Demand Pos Filtered [mm]
Time: 102.00 [ms/div] Trig: On Bit Up Pretrig: 20%

Trigger Mode: Trig on bit
Var: Trail Warning
Edge: Up
Drive: Drive A

Figure 5-20: Following error recording

In this example the point where the slider was blocked is clearly visible. The moment when the trigger is released is marked as zero. The prehistory of the following error can be recorded easily by means of the pretrigger function.

If the trigger condition “Edge” is set to “Down”, the trigger is only released when the following error disappears. You can record any variable, as for example “Current”. That way you can gain information on the force of the motor.

5.12.5 Final remark

These two examples were intended to give you an introduction in the operation of the Oscilloscope. In order to become even more accustomed with this tool, it is advisable to test different settings (for example recording and defining other curves and motors, or combinations with the other *LinMot*® Talk modules: Curve Inspector, Curve Creator and Parameter Inspector).

The knowledge of the possibilities and operational modes of the Oscilloscope can save you a good deal of time during testing and commissioning.

5.13 Tutorial C: Generating reference curves

In this example we want to create a reference curve by merging separate curve sections to build a single curve for a linear motor. We assume the following:

- The *LinMot*® electronic unit (Ex00-AT or Ex000-AT) is ON
- A linear motor LinMot Pxx-23 or Pxx-37 is connected to the "Mot A" port
- The *LinMot*® electronic unit is connected to the PC
- *LinMot*® Talk is running on the PC and the user has logged in

Goal

We create a curve that looks like the following:

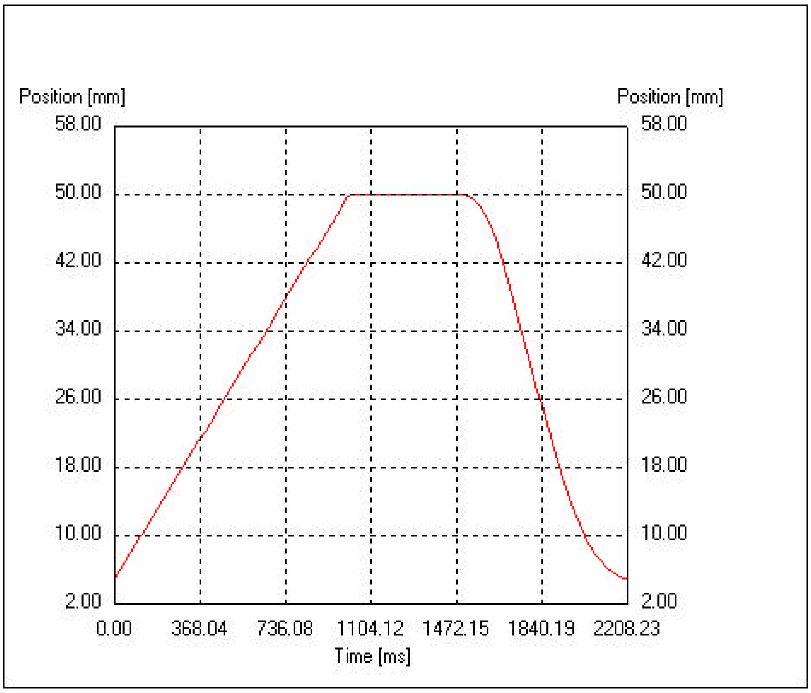
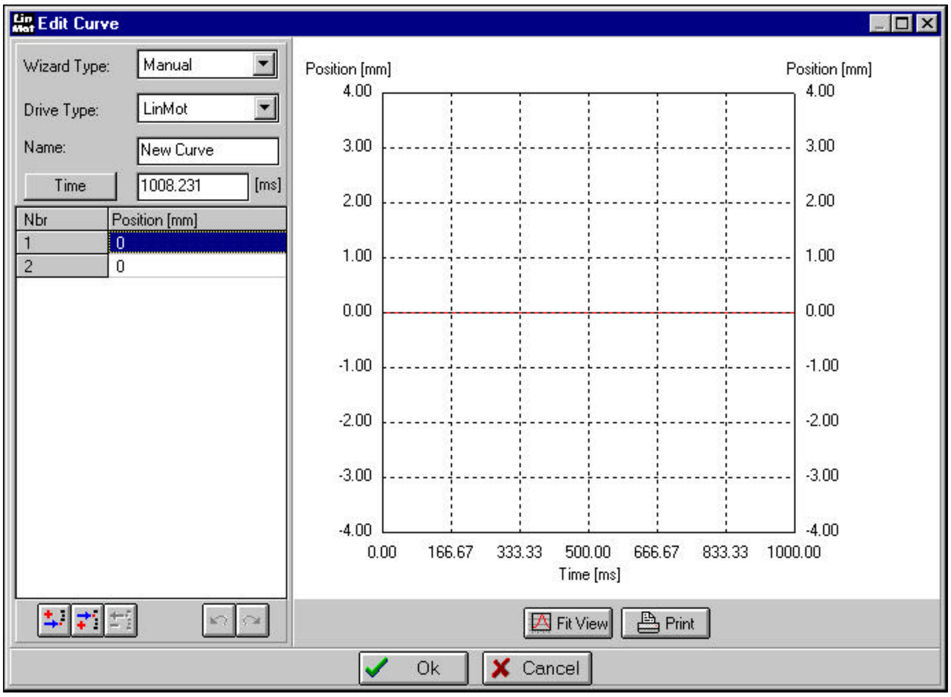


Figure 5-21: Goal of the example

The curve is built from three curve sections that we will first define separately. Afterwards we will join the three curves together and create our complete “work”. In the end we will let the curve be executed by the motor. No panic, this example proceeds in small steps and includes many figures that will help you visualize every step.

5.13.1 "Linear Out" curve

The first step consists in defining the linear movement outwards. For this purpose we open the Curve Inspector (by clicking on "Curve" in the Commander). With the "Create Curve" button we create a new curve that is displayed in the Curve Editor. The window should appear as follows:



Wizard Type: Manual
Drive Type: LinMot
Name: New Curve
Time 1008.231 [ms]

Nbr	Position [mm]
1	0
2	0

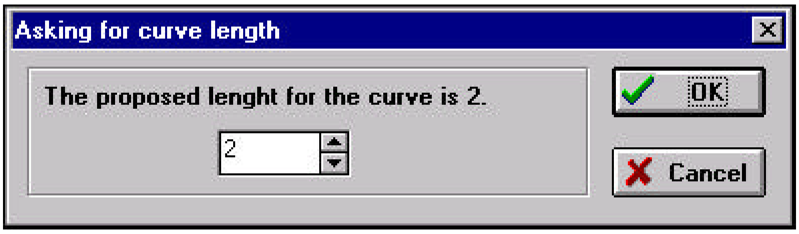
Figure 5-22: New curve in the Curve Editor

Now we must define our curve: first we shall select the type of curve generation. Under "Wizard Type" we select "Ramp" instead of the actually selected "Manual". The display changes and now includes the two fields "Start Point" and "End Point".

We now make sure that "LinMot Pxx-23" is selected as "Drive Type". If you wish to realize this example with a Pxx-37 choose "LinMot Pxx-37" as "Drive Type". In the following this alternative will only be shown between parenthesis.

We change the name to "Linear Out". At this point the actual movement sequence is defined. We want to create a curve that executes a linear movement from 5 mm to 50 mm in one second. For this purpose we enter 1000 [ms] in the "Time" field. If the unit is set to [Hz] it can be switched to [ms] with the "Frequency" button. After being entered, the value will be displayed as 1008 ms, as it is not possible for the electronic unit to meet the 1000 ms requirement exactly with just two points. In our case this causes no big problem. If this fact should cause a problem in a future application, you should increase the number of points.

We enter 5 [mm] for the “Start Position” and 50 [mm] for the “End Position”. The curve is now completely defined and we can click the “Calculate” button ▦ .
The following dialog box appears:


Asking for curve length

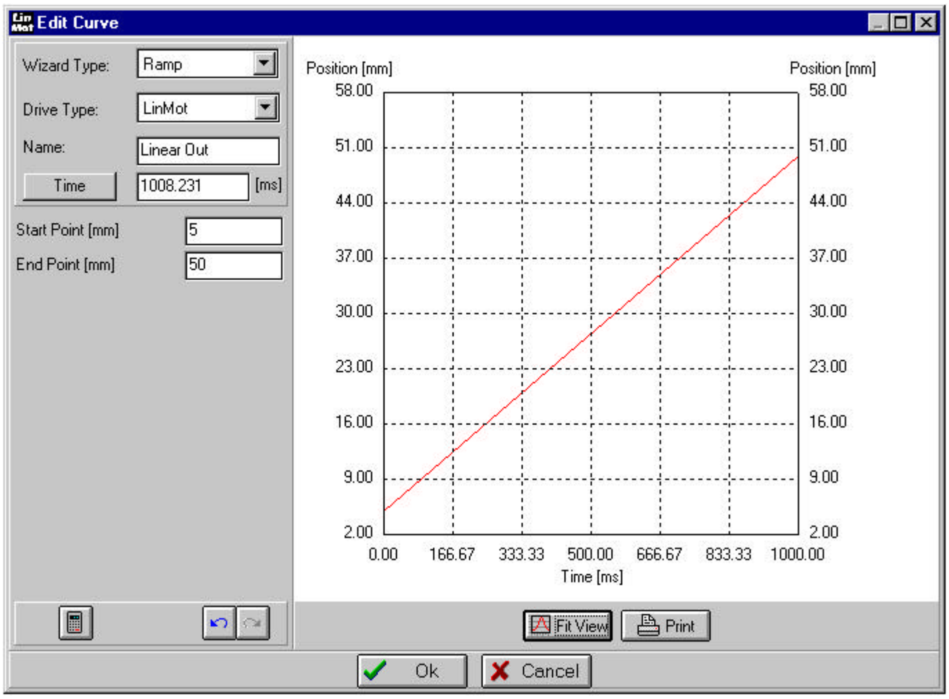
The proposed lenght for the curve is 2.

2

Figure 5-23: Dialog box for the number of points

The number of points can be increased at this stage in case the accuracy of the “Time” entry is insufficient for the application. We make do with two points and click on the “OK” button.

The defined curve should look like this:


Edit Curve

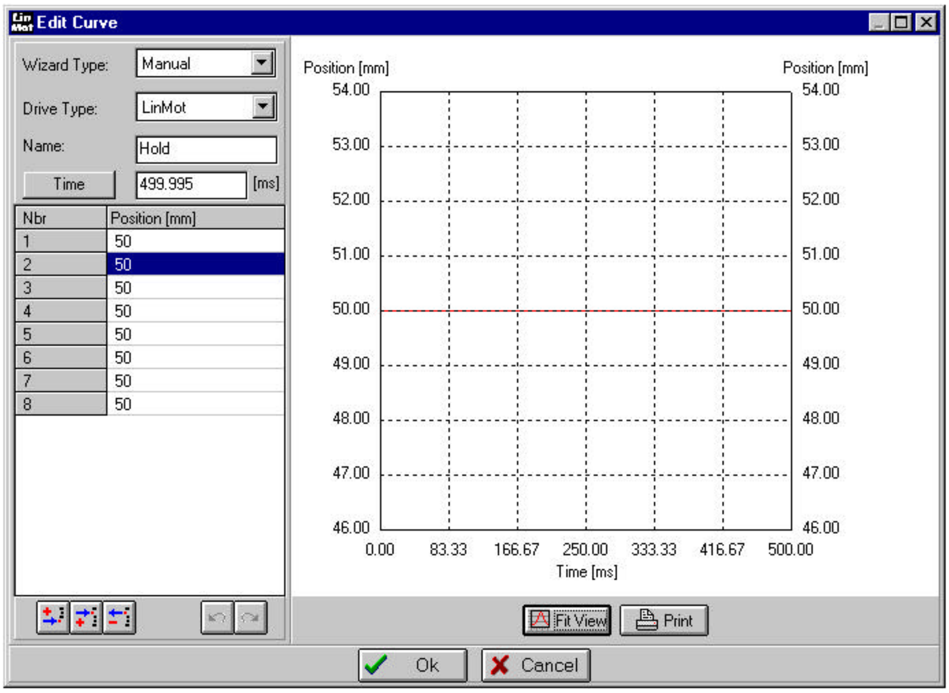
Wizard Type: Ramp
Drive Type: LinMot
Name: Linear Out
Time 1008.231 [ms]
Start Point [mm] 5
End Point [mm] 50

Figure 5-24: "Linear Out" curve

With the “OK” button we close the Curve Editor and see the curve “Linear Out” in the curve window of the Curve Inspector.

5.13.2 "Hold" curve

As next step we want to define the middle section of the curve, in which the motor holds the position at 50 mm during half a second. We open a new curve by clicking on the “Create Curve” button. We then enter the values in the Curve Editor. This step will be described only briefly. We set the wizard to “Manual”, the “Drive Type” to “LinMot" and enter “Hold” in the curve name field. We enter a time of 500 ms and twice 50 mm in the value fields. In order to increase the accuracy of the actual time we raise the number of points to 8. For this purpose we click on the “Add” button , enter the number 6 when asked and click on the “OK” button. The inserted points will automatically take the value 50 mm. The curve “Hold” is now finished and we see the following on the screen:



Edit Curve

Wizard Type: Manual

Drive Type: LinMot

Name: Hold

Time 499.995 [ms]

Nbr	Position [mm]
1	50
2	50
3	50
4	50
5	50
6	50
7	50
8	50

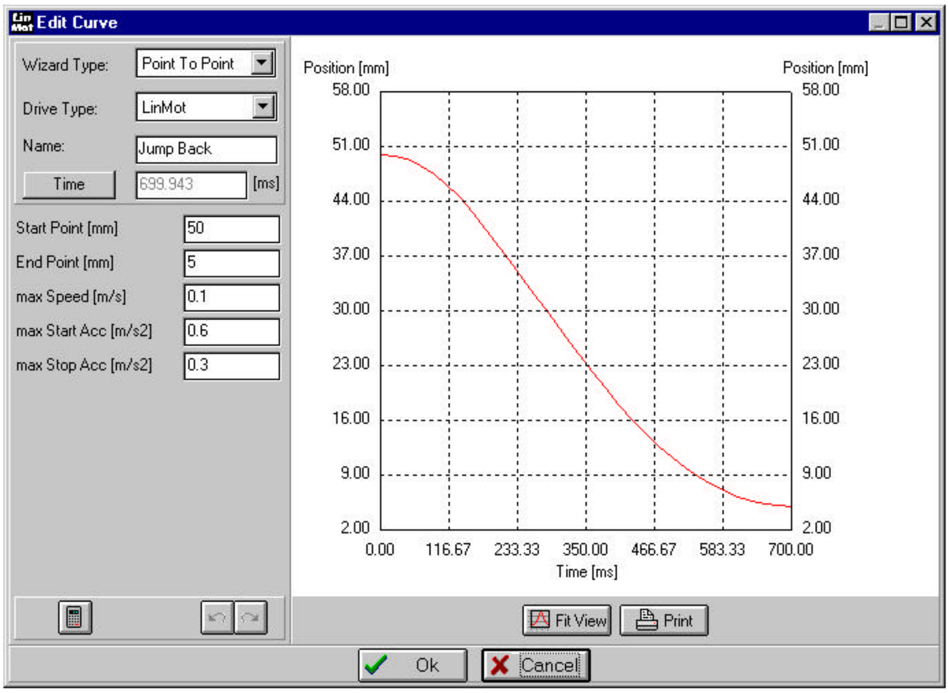
Figure 5-25: "Hold" curve

We close the Curve Editor with “OK”. We now have two curves in the Curve Inspector.

5.13.3 "Jump Back" curve

The last curve we define is the jump from 50 mm to 5 mm, which should be limited in speed and acceleration.

We open a new curve with “Create Curve” and set the “Wizard Type” to “Point to Point”, “Drive Type” to “LinMot”and name the curve to “Jump Back”. We cannot edit the time. The curve should bring the slider back, therefore we set the “Start Point” to 50 mm and the “End Point” to 5 mm. We then define the maximum allowed speed of 0.1 m/s. The maximum start and stop accelerations can be set separately. We set “max Start Acc” to 0.6 m/s^2 and “max Stop Acc” to 0.3 m/s^2. After clicking on the “Calculate” button the Curve Inspector suggests 32 points for the curve, which we accept by clicking “OK”. The time required for the execution of the curve is calculated automatically and amounts to 700 ms. The “Jump Back” curve look then like this:



Edit Curve

Wizard Type: Point To Point

Drive Type: LinMot

Name: Jump Back

Time 699.943 [ms]

Start Point [mm] 50

End Point [mm] 5

max Speed [m/s] 0.1

max Start Acc [m/s2] 0.6

max Stop Acc [m/s2] 0.3

Figure 5-26: Curve "Jump Back"

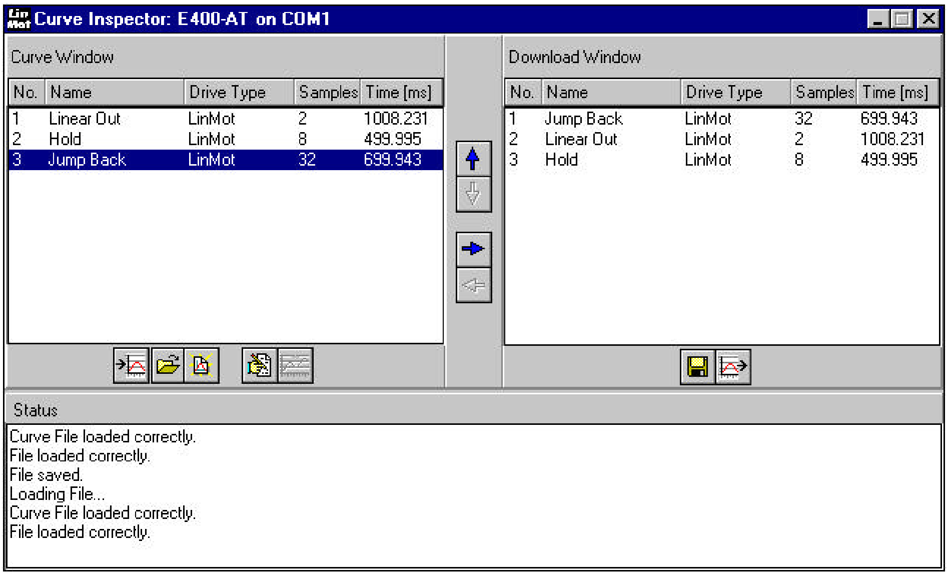
We close the Curve Editor with the “OK” button. The curve window of the Curve Inspector now lists the three curves we have defined.

5.13.4 Saving curves

In order not to loose the curves created, we want to save all three to a file. For this purpose we select them all with the mouse and click on the “Right” button . All the curves are now in the download window. We want the “Jump Back” curve to be the first in the list. For this we select the curve and click twice on the “Up” button . We now have the following sequence: 1. "Jump Back", 2. "Linear Out" and 3. "Hold". By clicking on the “Save” button a “Save file” dialog box opens, from which we can save the curves under the name “Curves1.csv”. After doing this we can open the file with the “Open” button . We now have the same curves in the left and right panels of the Curve Inspector window.

5.13.5 Merging curves

By saving a number of curves to a file the single curves remain separated. Should it be required to join the curves to a single, continuous movement, this can be done by means of the “Join” command. Before we do this, we must move the “Jump Back” curve back to the last position in the list (selecting “Jump Back” and clicking twice on “down”). The window should appear like this:



Curve Inspector: E400-AT on COM1

Curve Window

No.	Name	Drive Type	Samples	Time [ms]
1	Linear Out	LinMot	2	1008.231
2	Hold	LinMot	8	499.995
3	Jump Back	LinMot	32	699.943

Download Window

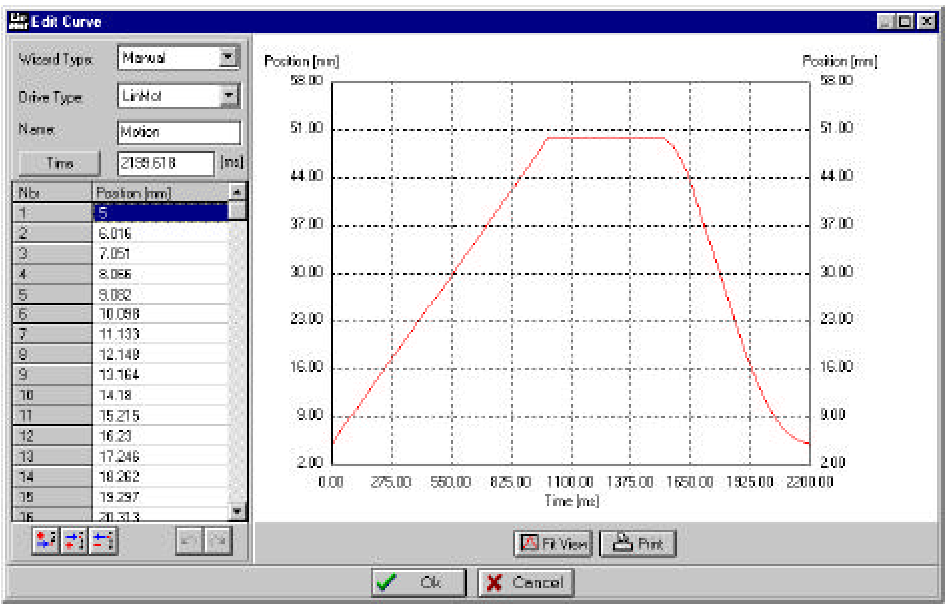
No.	Name	Drive Type	Samples	Time [ms]
1	Jump Back	LinMot	32	699.943
2	Linear Out	LinMot	2	1008.231
3	Hold	LinMot	8	499.995

Status

```
Curve File loaded correctly.
File loaded correctly.
File saved.
Loading File...
Curve File loaded correctly.
File loaded correctly.
```

Figure 5-27: Curve Inspector before merging the curves

Now to the merging: we select all the curves in the curve window and then click on the “Join” button . We accept the proposed number of points (98) with “OK”. The Curve Editor appears again. We name the curve “Motion” and see the following window:



Edit Curve

Wizard Type: Manual

Drive Type: LinMot

Name: Motion

Time: 2139.618 [ms]

Nbr	Position [mm]
1	5
2	6.016
3	7.051
4	8.066
5	9.082
6	10.098
7	11.133
8	12.148
9	13.164
10	14.18
11	15.215
12	16.23
13	17.246
14	18.262
15	19.297
16	20.313

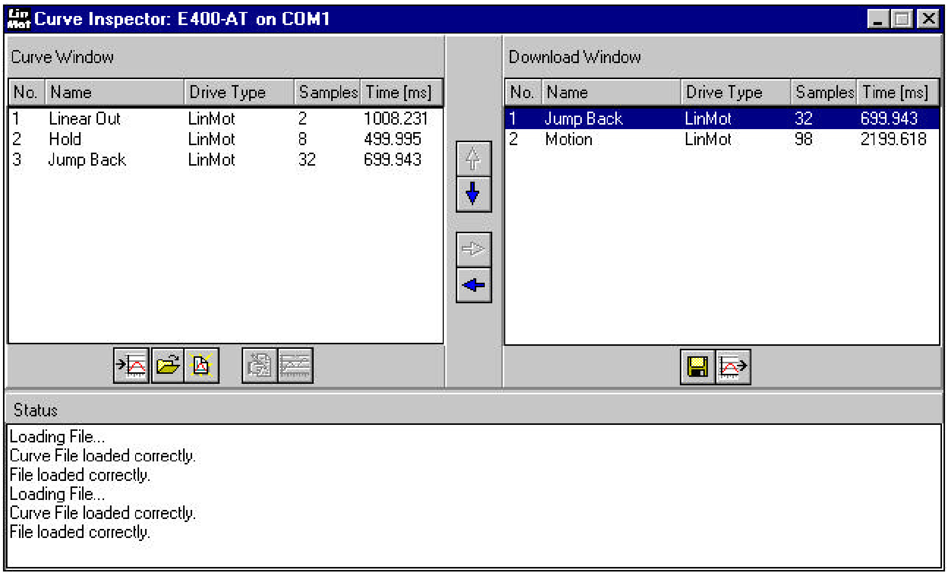
Figure 5-28: "Motion" curve

We close the Curve Editor with “OK”.

5.13.6 Writing curves to the electronic unit

As a last step we want to write the defined curves to the electronic unit, so that we can see the actual movement be executed by the motor.

We select the “Motion” curve in the curve window and move it to the download window by clicking on the “Right” button. There are now four curves in the download window. We delete the curves “Linear Out” and “Hold” by selecting them and pressing the “Delete” key on the keyboard. The Curve Inspector window now appears as follows:



Curve Inspector: E400-AT on COM1

Curve Window

No.	Name	Drive Type	Samples	Time [ms]
1	Linear Out	LinMot	2	1008.231
2	Hold	LinMot	8	499.995
3	Jump Back	LinMot	32	699.943

Download Window

No.	Name	Drive Type	Samples	Time [ms]
1	Jump Back	LinMot	32	699.943
2	Motion	LinMot	98	2199.618

Status

```
Loading File...
Curve File loaded correctly.
File loaded correctly.
Loading File...
Curve File loaded correctly.
File loaded correctly.
```

Figure 5-29: The Curve Inspector before the download

We save the two curves that are in the download window. For this purpose we click on the “Save” button and give the file the name “Motion1.csv”. We now program the two curves in the electronic unit. This is done by clicking on the “Download” button and confirming the process with “OK”. The curves are now being written to the electronic unit.

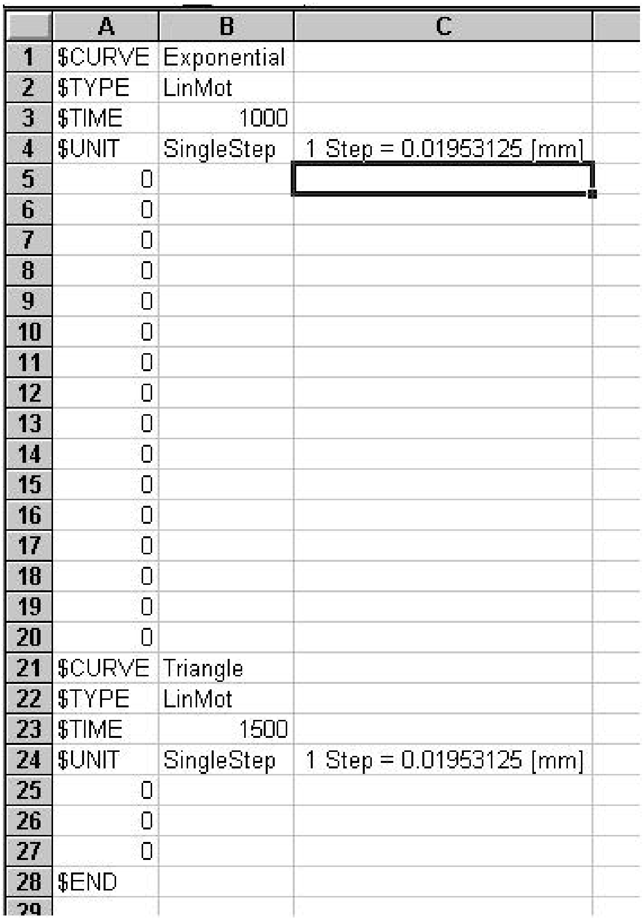
Finally, we test the curves. For this the electronic unit must be configured correctly. We load the standard configuration by clicking on “Import” and selecting the one appropriate for the electronic unit we are using. For an E400 this would be 400AT01.cfg, for an E1000 it would be 1000AT01.cfg and so on. The configuration is now loaded and we can adjust the parameters according to our application.

We set the “Drive Type” of channel A to LinMot Pxx-23 (LinMot Pxx-37) in Drives\Drive A\Type. As the curve we created was written to the electronic unit as number 2, we must set Drives\Drive A\Set Value Generation\Curve Number to 2. We are now ready to start. We open the Control Panel (“Control” button in the Commander), click on “Stop” and then on “Start”. The motor initializes and executes our “Motion” curve cyclically.

5.14 Tutorial D: Defining curves with Excel

In this chapter we take a look at how to define and edit curves with Excel. It is assumed that you are already accustomed to using the Curve Inspector and the Curve Editor.

With this example, in which we want to define an exponential curve and a triangle curve, we will explain the necessary steps. With the Curve Inspector, res. Curve Editor, we will first generate a template for the curve file, so you don’t have to build the file itself. For this purpose we create two new curves, one named “Exponential” with 16 points and 1000 ms as time and the second one named “Triangle”, with 3 points and 1500 ms. Both curves are created with the “Manual” wizard, and the motor selected should be “LinMotPxx-23” or “LinMotPxx-37”, depending on what you are using. These two curves are then saved together in a file named “ExelTest.csv”. At this point we start Excel and open the file “ExelTest.csv”. The following table appears:



	A	B	C
1	$CURVE	Exponential	
2	$TYPE	LinMot	
3	$TIME	1000	
4	$UNIT	SingleStep	1 Step = 0.01953125 [mm]
5	0		
6	0		
7	0		
8	0		
9	0		
10	0		
11	0		
12	0		
13	0		
14	0		
15	0		
16	0		
17	0		
18	0		
19	0		
20	0		
21	$CURVE	Triangle	
22	$TYPE	LinMot	
23	$TIME	1500	
24	$UNIT	SingleStep	1 Step = 0.01953125 [mm]
25	0		
26	0		
27	0		
28	$END		
29			

Figure 5-30: Starting point in EXCEL

We see both curves with their definitions and points, all set to zero, in column A. We can now insert formulas or references. In doing so we must notice that the values must be in column A and must be defined as “SingleStep”. Decimal points are not allowed.

We will edit the “Exponential” curve, so that an exponential jump from 5 to 50 mm is executed. For this we define the exponents in column C with the values 0, 0.3, 0.6, ... 4.5. In column B we enter the following formula

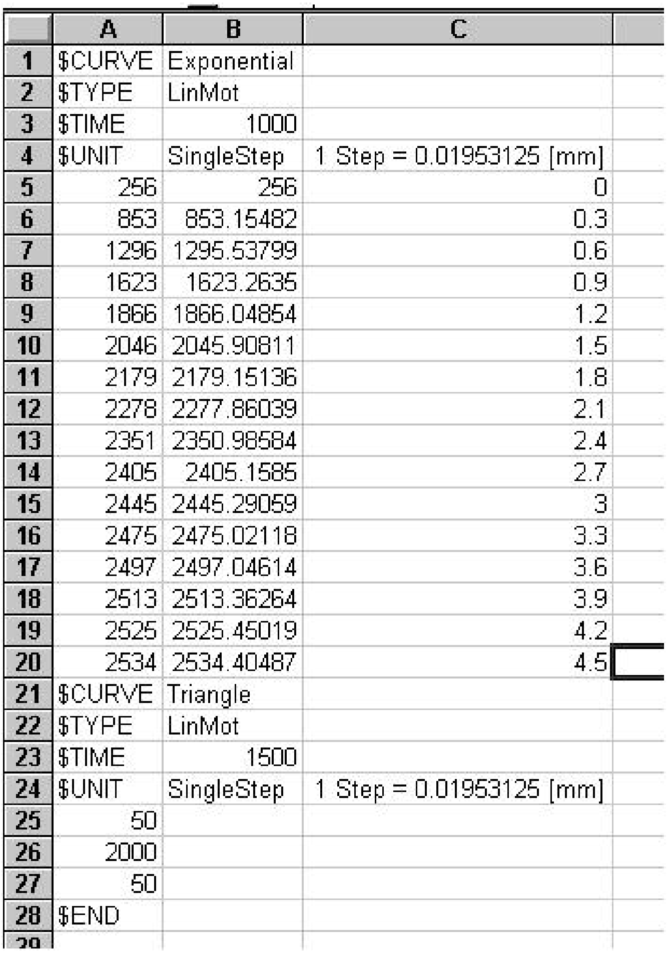
Bn = 1024/20*(45*(1-1/EXP(Cn))+5)

for the definition of the values. The factor 1024/20 is used for scaling from millimeters to Singlesteps. In column A we enter the following function

An = ROUND(Bn,0)

in order to eliminate the decimal point. With this the “Exponential” curve is complete.

For the “Triangle” curve we enter the values 50, 2000 and 50 manually. You should now see the following table:

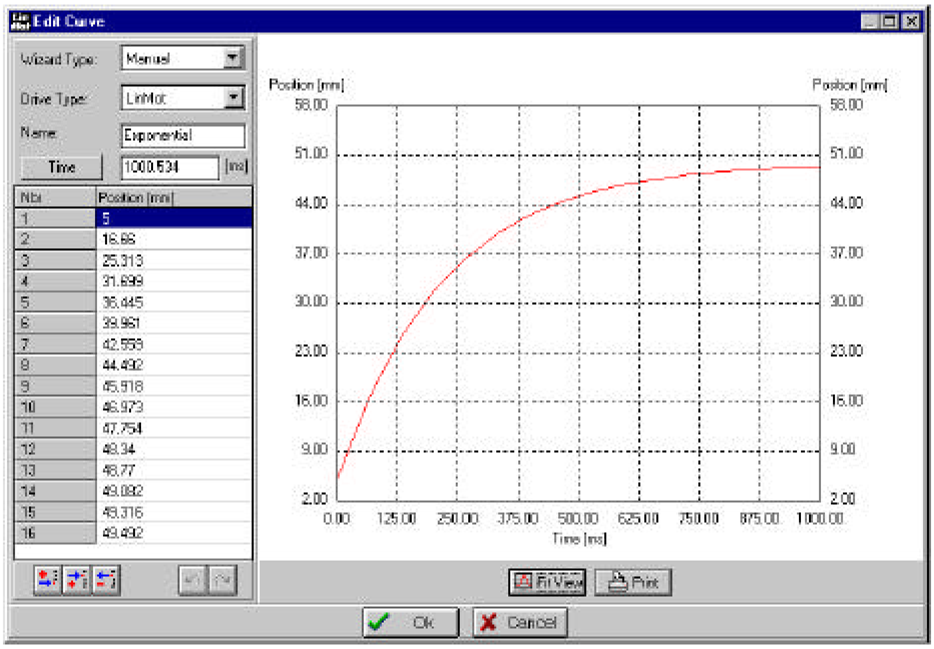


	A	B	C
1	$CURVE	Exponential	
2	$TYPE	LinMot	
3	$TIME	1000	
4	$UNIT	SingleStep	1 Step = 0.01953125 [mm]
5	256	256	0
6	853	853.15482	0.3
7	1296	1295.53799	0.6
8	1623	1623.2635	0.9
9	1866	1866.04854	1.2
10	2046	2045.90811	1.5
11	2179	2179.15136	1.8
12	2278	2277.86039	2.1
13	2351	2350.98584	2.4
14	2405	2405.1585	2.7
15	2445	2445.29059	3
16	2475	2475.02118	3.3
17	2497	2497.04614	3.6
18	2513	2513.36264	3.9
19	2525	2525.45019	4.2
20	2534	2534.40487	4.5
21	$CURVE	Triangle	
22	$TYPE	LinMot	
23	$TIME	1500	
24	$UNIT	SingleStep	1 Step = 0.01953125 [mm]
25	50		
26	2000		
27	50		
28	$END		

Figure 5-31: The curves defined in Excel

In order not to loose the formulas in the table it is important to save the table as *.xls. We save the document as “ExelTest.xls” first, and then as “ExelTest.csv”, which is the format readable by the curve inspector. It is important to select the file type under “Save As” in Excel, and not by manually entering the file extension.

We now load the file in the Curve Inspector (ExelTest.csv) and take a look at the curve, that should appear as follows:



Wizard Type: Manuel

Drive Type: LinMot

Name: Exponential

Time 1000.534 [ms]

Nbr	Position [mm]
1	5
2	16.66
3	25.313
4	31.699
5	36.445
6	39.961
7	42.559
8	44.492
9	45.918
10	46.973
11	47.754
12	48.34
13	48.77
14	49.082
15	49.316
16	49.492

Figure 5-32: “Exponential” curve

There are many other possibilities in Excel. Some are listed here briefly:

- Changing the number of points in a curve by inserting or deleting
- Calculating the time with formulas
- Renaming curves
- Changing motor type
- Inserting or deleting entire curves
- Defining and calculating complex curves with special macros (ev. interactive)

It is important that the structure of the file remains unchanged (curve header, values and the keyword “$END” after the last curve).

6. Parameters

Refer to User Manual for SW Rel. 1.3.

7. Service

Refer to User Manual for SW. Rel. 1.3.