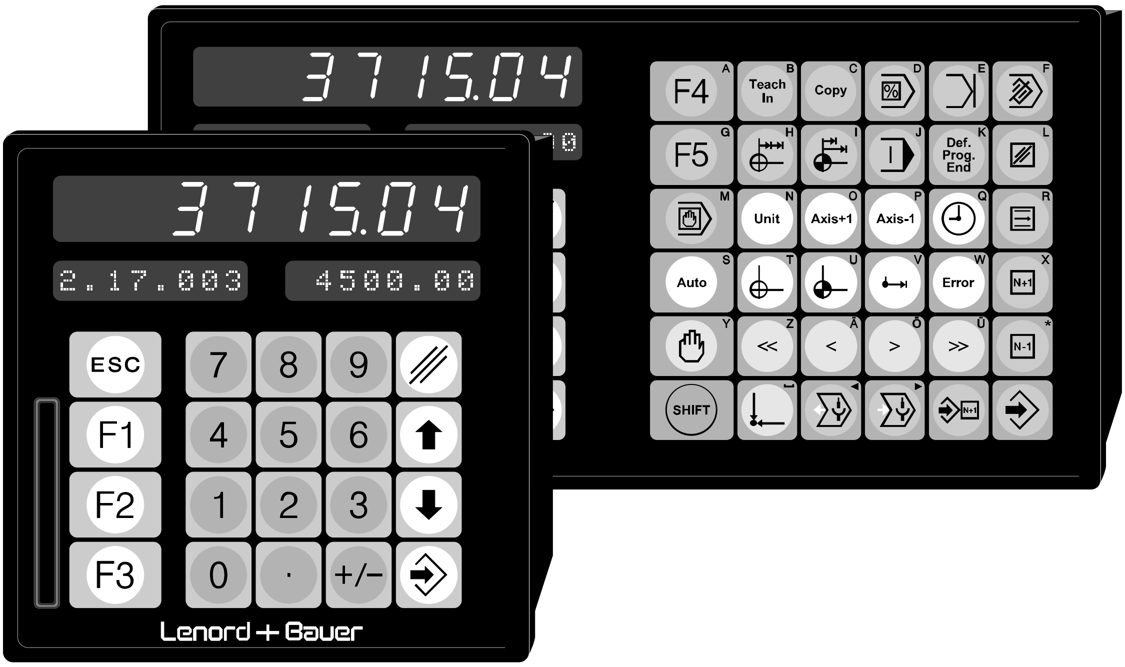
Operating Manual

Positioning Controller

GEL 8310 / 8610

GEL 8610 **for 1…6-axis applications**



GEL 8310 **for 1…3-axis applications**

Lenord + Bauer

 Subject to change without notice

Lenord + Bauer
Lenord, Bauer & Co. GmbH
Dohlenstraße 32
D - 46 145 Oberhausen
Telefon (0208) 99 63 - 0 • Fax (0208) 67 62 92
Internet: http://www.lenord.de • E-Mail: info@lenord.de

Trouble shooting 6

Mounting instructions 7

Storage locations for machine parameters Appendix A

Pin Layout Appendix B

1 Introduction

1.1 General safety instructions

▲ The Controllers GEL 8310/8610 have been built according to state-of-the-art-technique and in compliance with valid safety regulations. However, upon using them it might happen that there will be the danger of injury for the user or other persons resp. damages to the Controller itself or other material assets. Do use the Controller only

- for the designated use (see section 1.2)
- in perfect technical condition.

In order to maintain this condition and to ensure operation free from danger, installation, wiring and service work should be carried out only by **qualified specialists**[1], under observance of the relevant accident prevention, safety guidelines and information in the product documentation.

▲ If through a failure or fault of the Controller an **endangering** of persons or **damage** to plant is possible, this must be prevented using additional safety measures (E-STOP, limit switches etc.). These must remain operational in all Controller operating modes, so that through disengaging of the safety features no **uncontrolled re-start** of the controlled machine is possible.

▲ If **danger-free operation** can no longer be ensured, the Controller should be taken out of commission and secured against accidental operation.

▲ When servicing make sure that there is **no voltage** supplied to the **whole** controller, i.e. including the built in module cards. This applies, in particular, to **relay cards** used which are operated at high voltages (up to **240 VAC**)! When changing modules ensure that good earthing has been performed in the environment to avoid electrostatic charges.

▲ Necessary repairs to the Controller are only to be carried out by LENORD+BAUER or a specifically authorised agent.

Note: Some of the auxiliary voltage outputs on several modules are equipped with self-resetting thermal fuses (PTC overload protection, see appendix B). In case of release, switch off the voltage supply for approx. 20 seconds.

[1] These are persons, who
- in respect of the project are familiar with the safety concepts of automation technology,
- are experienced in the area of EMC,
- have received training in relation to installation and service work,
- are familiar with the operation of the unit and know the pertinent product information contained to ensure faultless and safe operation.

Concerning the auxiliary voltages, please read the instructions at the beginning of the section ‘Connection diagrams’ in appendix B.

▲ The operating data tolerances for the supply voltage stated in appendix C must be strictly observed, since otherwise Controller **functions might fail** causing **dangerous situations** (the mains failure monitoring system in the Controller reacts and a started programme might possibly not be continued properly). If required, provide for an E-STOP circuit reacting to this.

▲ Prior to start carrying out any **positioning procedures** using the Controller, it is imperative that you read section **4.1.3** and the instructions on commissioning supplied in chapter 5 (especially sections **5.3** and **5.4**).

▲ Further special **information on danger** causing injury or material damages and any other information are marked in this manual by special symbols:

DANGER

This symbol denotes an **imminent danger** to life and health of persons. *Should you ignore these warnings severe damages to health or even life-threatening injuries might be the consequence.*

WARNING

Symbolises a **possible danger** for life and health of a person. *Should you ignore these warnings severe damages to health or even life-threatening injuries might be the consequence – or may cause severe material damages.*

CAUTION

This symbol denotes a **possibly dangerous situation**. *Should you ignore these warnings slight injuries or material damages might be the consequence.*

ATTENTION

This symbol denotes in general **critical situations** or **possible material damages**.

Where you see this symbol you will be supplied important information for the **proper handling** of the Controller. *Should you ignore this information a malfunction might occur in the Controller or the machines connected with the Controller..*

Tip

Here you are supplied with **application hints** and other useful information for the optimum use of the Controller.

1.2 Designated use

The Controllers type GEL 8310 and GEL 8610 are designated to control and regulate drives in industrial and commercial areas. They may only be operated in built-in condition.

Special options and devices made to customers' specifications may result in an extension resp. restriction of the above prescription. Should this be the case, the specification given in the pertaining descriptions are binding.

Designated use also encompasses that the instructions in this user manual are followed.

A differing or exceeding use is deemed not designated. In such cases LENORD, BAUER & CO. GMBH does not accept any responsibility for damages.

1.3 Guarantee, liability and copyright

Fundamentally our general delivery and payment conditions apply, which are available to the user at the latest upon closing contracts. **Guarantee and liability claims** on damages to persons or objects are excluded, if they can be traced to one or more of the following causes:

- implementation of the Controller outside the designated use
- inappropriate mounting, installation and operation of the Controller
- operation of the Controller in conjunction with defective or non functional safety equipment in the system
- ignoring the instructions in the user manual with regard to storage, mounting, installation and operation of the Controller
- arbitrary build changes to the Controller
- improper repairs
- catastrophes due to foreign bodies and higher forces

The user manual has been produced with great care. However no guarantees can be made for possible errors.

1.4 Information on this manual

📖 This manual is applicable to controllers with the **standard software**

version **14.00** and higher

Appendix **Z** supplies information as to whether the **software was updated** (see also the last item of this section).

Does a Controller not comply with the indicated version, the contents may not be regarded as binding. We cannot assume any **liability** for any malfunction resulting thereof and its consequences.

📖 Due to the Controller's universality and the multitude of its functions the user manual becomes automatically **more extensive**, since all possible versions and variations must be considered.

The concept of the present manual is based on **selective reading**, i.e., you are not supposed to completely study each chapter in order to be able to use the Controller most efficiently. This is the reason why – because of the absence of an index – the **table of contents** at the beginning of the manual and of each chapter is very detailed thus facilitating the **search** for a specific information.

Optional functions are marked as such so that you may ignore this information right from the start, if your Controller is not equipped with these special functions.

For **orientation** purposes, the individual chapters are briefly described below:

- chapter 1: Safety instructions and general information
  To be studied by all means!

- chapter 2: Brief description survey chart of the operating elements
  should you operate the Controller via the optional PC program BB8110, please refer to the respective manual which is supplied separately

- chapter 3: Survey and description of possible modes and states of operation of the Controller as well as the corresponding applicable functions and keys
  provides important information especially for familiarizing purposes

- chapter 4: Detailed functional descriptions
  is first and foremost intended to serve as a kind of 'reference book'; however, do not fail to read the first section 'Definitions'

- chapter 5: How to **commission** the Controllers

  though there are quite a number of techniques for the commissioning of the Controller depending on individual experiences, please ***do not fail to consider*** *the information supplied in this chapter*

- chapter 6: The most important warning and error messages shown in form of a chart indicating causes and how to put things right (option messages are mentioned where they are described: appendix O)

- chapter 7: Instructions on performing cable connections, exchanging PCBs (modules) and replacing an EPROM

  please note in particular the EMC-relevant measures

- appendix A: A list of all storage locations for machine parameters including brief descriptions – in tabular form

  this survey might be sufficient for the experienced user for programming the Controller

- appendix B: Information on the different Controller modules and their pin layout

- appendix C: Technical data, dimensioned drawings and type codes

- appendix D, E, ...: Descriptions of software upgrades or modifications (only available, if required)

- appendix O: Description(s) of employed, i.e. implemented options such as ‘circular interpolation’ or ‘synchronisation control’ (GEL 8910.xxxx)

- appendix Y: A collection of forms and supplementary sheets (if required)

- appendix Z: A collection of updating sheets for software upgrades or necessary corrections (if available; see also last point of this paragraph)

📖 In general, the information given in this operating manual refers to positioning controllers with a **maximum possible number of axes** (3 axes for the GEL 8310 or 6 axes for the GEL 8610, respectively). If equipment with less axes is used, mind the respective restrictions in the designs (e.g. for the programming of machine parameters in appendix A).

📖 In case of **terminal designations** (e.g. Z8) the letter denotes the terminal strip or connector and the number the pertaining terminal or pin (see appendix B). If several identical connectors exist, an index is added between the letter and the terminal no. (e.g. $A_2$10 for pin 10 of the 2nd connector A [A_2]). If not expressly mentioned, all data listed apply to the standard device resp. the standard pin layout of the GEL 8310/8610.

- **Numbers of storage locations** of machine parameters do have a prefix referring to the respective parameter plane (see appendix A):

  1/x = system parameter x **2/**x = unit parameter x
  3/x = axis parameter x

- A '**x**' or '**X**' in a numerical expression represents any numeral from the admissible range of values for this expression (example: GEL 8x10 = GEL 8310 and/or GEL 8610).

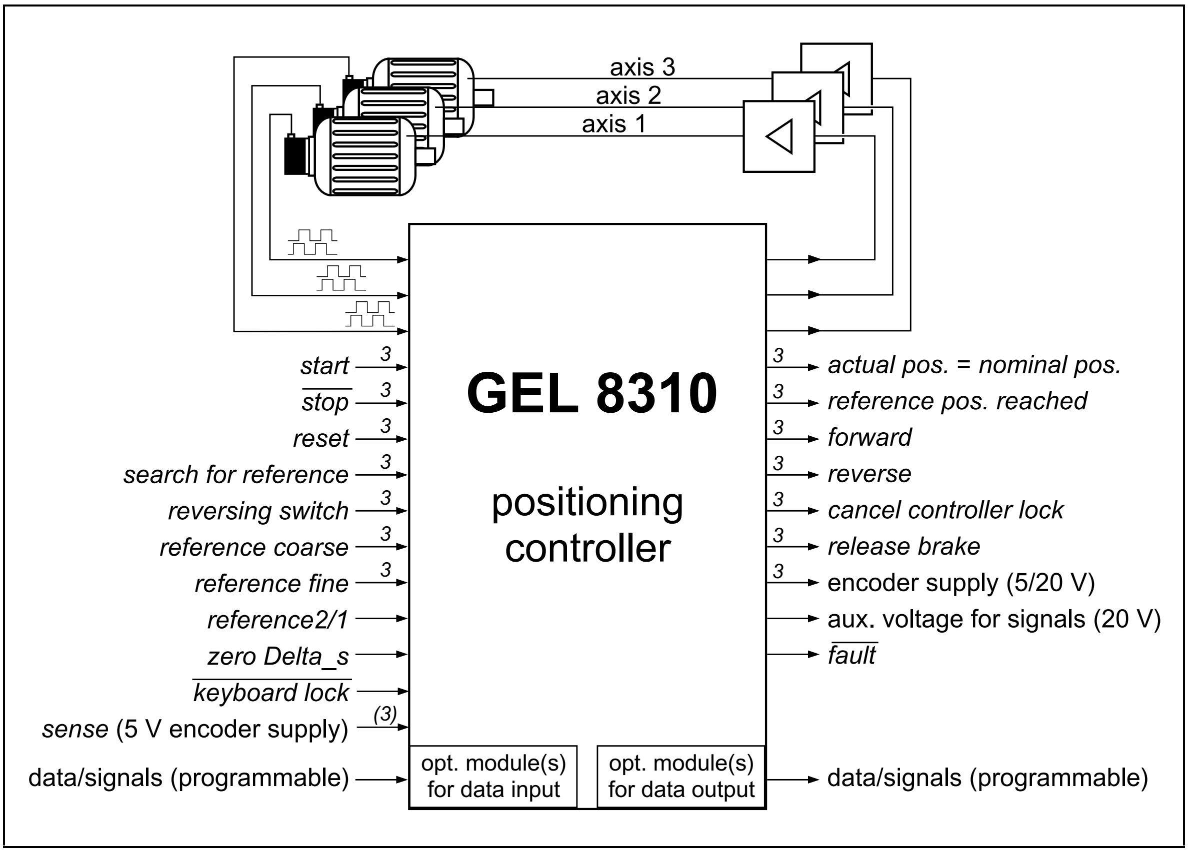
- Texts printed in *italics* normally describe **input or output signals** at the terminal strips (see appendix B).

- If **functions are added**, which normally means that the software version will be given a new number, new descriptions will either be attached in form of exchange sheets or as separate annexes (appendix D and following). In both cases, and also if corrections become necessary, an updating sheet will be supplied to be added to appendix Z in order to inform about the current software state of the manual.

1.5 Characteristics of the Controller GEL 8310/8610

- The menu oriented operating/programming is performed using plain text either via
  - the keyboard of the Controller or
  - the optional PC program BB8110
- single drives (axes) may be controlled separately or together as a so-called unit
- exact, dynamic close-loop control with speed pre-control
- very short control sampling time of 1 ms per connected axis
- path control is possible (optional with circular interpolation)
- sequential storage of malfunctions
- sentence structure (nominal value types) can be set differently for each unit (combination of axes); see para. 3.2
- all storage locations are protected against mains failure (EEPROM with a warranted service life of 10 years or 10,000 rewrites)
- possible data storage on an external data carrier (memory card, option)
- optional extension of functions, e.g. synchronisation control or control for a flying saw
- maintenance-free (clean the housing with a wet cloth)

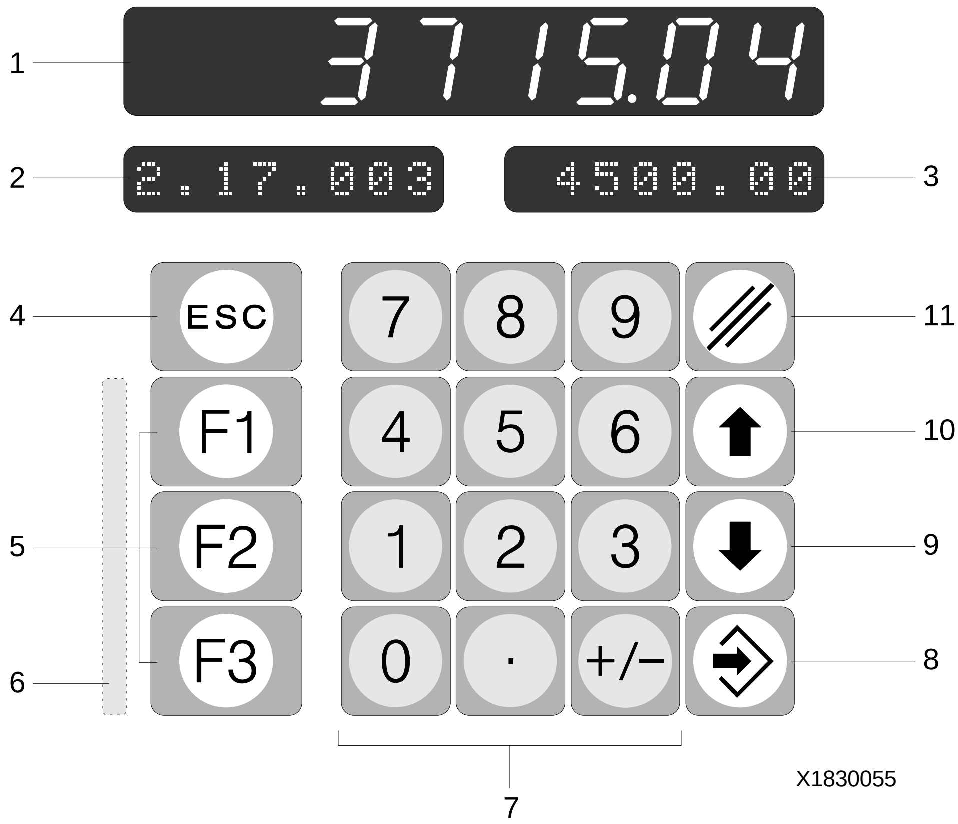
The following picture shows the basic structure of a 3-axes positioning device with the GEL 8310 Controller and incremental encoders, including the possible input and output signals.
Depending on the software version available and the extension of functions other signals as well might be available. For further information, please refer to the pin layout in appendix B, the description of possible options (Annex O), or to possibly existing upgrading annexes (appendix D, E, ...).



E1831014

2 Operational Controls

2.1 Basic keyboard



1 **Display A**: actual value counter, displays the input of nominal values and machine data

2 **Display B**: active operating parameters, plain text

3 **Display C**: nominal/actual values, plain text

4 **ESCAPE key**: cancel a function and return to the next higher functional level

5 **Function keys**: activation of a certain function in combination with another key (see section 2.3)

6 **Slot for memory card** (option): to read or write operating data

7 **Data keys**: to input values in the programming mode

8 **ENTER key**: storing of an entered value or operating function, confirming messages of the Controller

9 **FORWARD cursor key**: scrolling forward through value and function lists

10 **REVERSE cursor key**: scrolling reverse through value and function lists

11 **CLEAR key**: resetting of a programmed or entered value to zero

2.2 Additional keyboard of the GEL 8610

A, G	function keys (reserved)
B	switch on TEACH-IN operation
C	copy sentence(s)
D, E	jump to the beginning/end of the program
F	delete program (with security inquiry)
H	incremental dimensions
I	absolute dimensions
J	continuous sentence processing
K	define the end of program
L	delete sentence
M	programming mode for nominal values
N	select unit
O, P	increment/decrement axis no.
Q	inquire date&time of error occurrence (option)
R	insert sentence
S	automatic mode
T, U	enter/search reference measure
V	enter correction value
W	call up fault memory
X, *	go to the beginning of the next/previous sentence
Y	manual operation (in combination with [Z], [Ä], [Ö], [Ü], [␣])
Z, Ä	fast/slow speed reverse
Ö, Ü	slow/fast speed forward
SHIFT	switch-over to capital letters for plain text input
␣	output *actual=nominal* signal; blank character for plain text input
◀, ▶	read/write memory card; cursor left/right for plain text input
a	store and go to next nominal value of same type
b	store; confirm messages

2.3 Function key combinations

The key combinations listed below are related to the Controller standard version (valid at the time of printing of this manual). If there are one or more functional options inserted in the Controller you will find more information in the special description of the option (appendix O).

More detailed information on the listed functions and the key equivalents of the GEL 8610 you will find in chapter 3.

F1	mode [1]	function
–	M	increment variant
·	A	display time (with serial interface with clock only)
0	A	list stored faults
1	A	enter a reference value
1	N	absolute coordinates offset
2	N	relative coordinates offset
2	A	search for reference
3	A	enter correction value
4	A	enter value of park position
9	N	Teach-in operation ON (ESC = OFF)
⬆	A	slow speed reverse
⬇	A	slow speed forward
⬆	N	beginning of previous sentence
⬇	N	beginning of next sentence
➡	N	store and go to the next nominal value of the same type

F2	mode [1]	function
0	N, M	return to automatic mode
0	A	enter the programming mode for nominal values
1	A	enter the programming mode for machine parameters
8	M	copy axis parameters

[1] valid mode: A = **a**utomatic mode, N = programming mode for **n**ominal values; M = programming mode for **m**achine parameters

F2	mode [1]	function
⬆	A	fast speed reverse
⬇	A	fast speed forward
➡	A	write to memory card (option)

F3	mode [1]	function
–	M	decrement variant
·	A	display number of actual software version
+/–	A	display signal states (password 9320)
0	A	select unit for display
0	N	floating or fixed zero processing (position ↔ length)
0	N	continuous sentence processing (speed)
1	A	select program
1	N	program flow instruction »CALL Pr.«
2	N	program flow instruction »JUMP Pr.«
3	N	program flow instruction »JMP Sent«
4	N	program flow instruction »IF I/O«
7	M, N	clear memory, delete unit or program
8	N	copy sentences
9	N	define the end of program
///	N	delete sentence
⬆	A	decrement axis for display (within the unit)
⬇	A	increment axis for display (within the unit)
⬆	N	jump to the beginning of program
⬇	N	jump to the end of program
➡	N	insert sentence
➡	A	read memory card (option)

[1] valid mode: A = **a**utomatic mode, N = programming mode for **n**ominal values;
M = programming mode for **m**achine parameters

3 Operating Modes and States

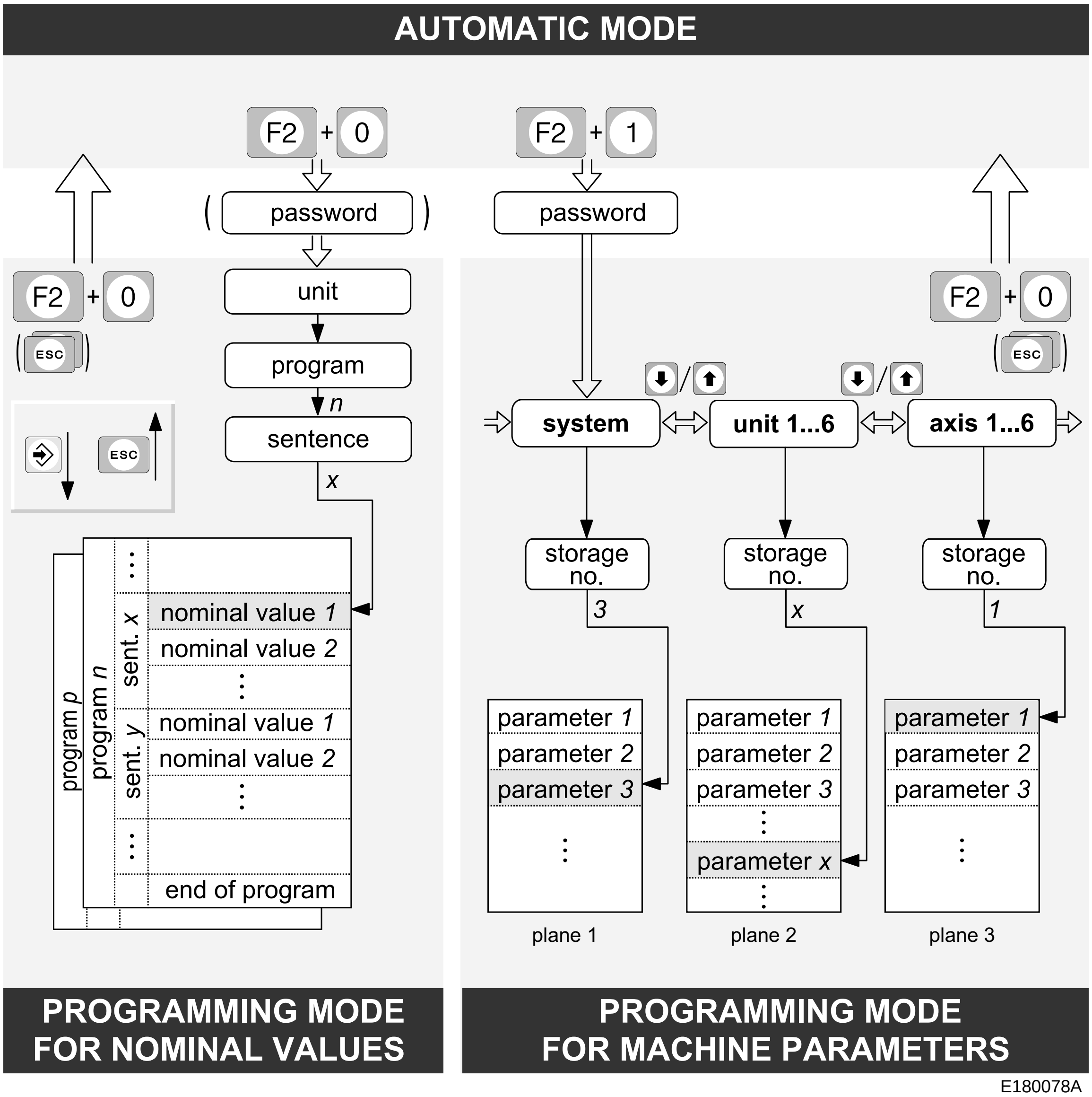
The Controller is designed for three operating modes and three operating states:

❑ **Automatic mode**

- started state
- interrupted state (stop)
- reset state

❑ **Programming mode for nominal values**

❑ **Programming mode for machine parameters**



3.1 Automatic mode

This is the *normal* operating mode of the Controller. The nominal/actual values of the active sentence may be read in displays B and C by pressing the cursor keys ⬇ and ⬆ (refer to section 3.1.2).

If display B only shows a **point** the power supply falls short of one of the minimum values specified at the operational data in appendix C.

The 3 possible operating states can be activated separately for each configured unit through appropriate signals at terminal strip **P** or via the serial interface (PC program BB8110).

Started state

In this state a defined nominal value program is processed.

Precondition:
- **High** level is applied at the **/stop** input of the respective unit (terminal P4/P7/P10)
- **Low** level is applied at the **reset** input of the respective unit (terminal P5/P8/P11)

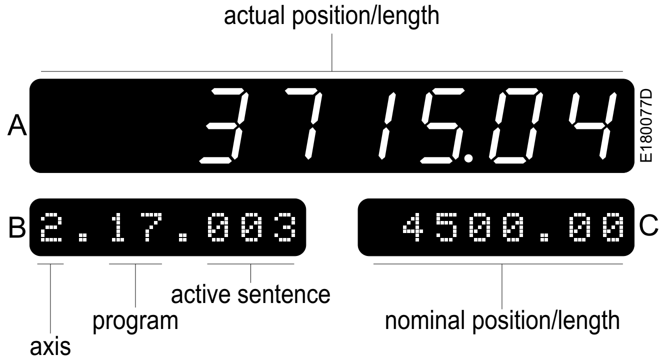
Activation:
- **positive** signal **edge** at **start** input P3/P6/P9
- via a serial interface

With each further start signal the next sentence of the program or another piece will be processed.

Within the started state further start signals may also be generated automatically i. e. internally:

- with continuous sentence processing (cf. section 4.3)
- with program flow instructions or coordinates offset (cf. sections 4.14 and 4.15)
- after a preset time has elapsed after the *actual=nominal* signal
  - identically for all sentences within the unit (cf. storage location 2/**10**)
  - differently for individual sentences within the unit (nominal value preset, cf. storage location 2/**1**)

Standard display:



Interrupted state (stop)

Here the program processing was stopped (temporarily) and can be resumed by the next start signal. The sentence number shown in display B flashes.

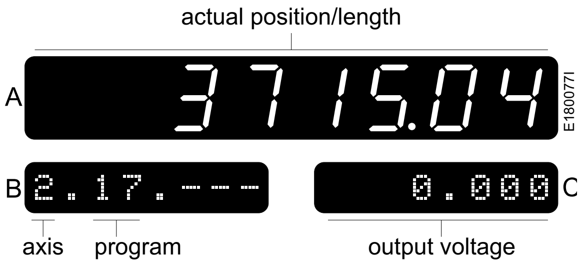
Activation:
- **Low** level at **/stop** input P4/P7/P10
- maximum permissible trailing distance has been exceeded (cf. storage locations 3/**42** and 3/**43** in appendix A) or a limit switch has triggered (cf. section 4.11); then, additionally, the ***/fault*** signal is active and the **separation points** in display B are **flashing** (cf. section 4.5.2)
- via a serial interface

Reset state

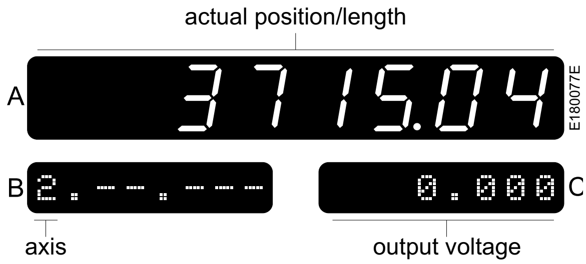
In this state several direct entries or an automatic search routine may be performed (described in the following section). Only if the Controller is in the reset state for all units, data can be read from the optional **memory card**.

Activation:
- **High** level at **reset** input P5/P8/P11
- selecting a storage location in the programming mode of machine parameters
- via a serial interface

Standard display:



program selection via keyboard



program selection via data input module

Instead of the output voltage, the contouring error or the actual positions of the other axis (axes) may be displayed, depending on which display mode has been selected in the reset state before.

3.1.1 Functions

The following keys and key combinations are available in the automatic mode to call up certain functions:

[↑], [↓]
Scroll the different **actual/nominal values** in displays B and C (refer to section 3.1.2);
the number of possibilities is determined by the programming of the machine parameters for the respective unit (sentence structure)

[ESC]
Reset display B to standard display ('axis.program.sentence'), display C does not change;

once a cursor key is pressed, display B will show again the identification text of the value displayed in C (cf. section 3.1.2)

[F3] + [1]
Select a program (only possible in the reset state and if the appropriate unit parameter 2/**6** = 0, »keyboard«)

[F3] + [0]
[Unit] N
Select a unit for the display;

if the unit includes more than one axis the one with the lowest number will be shown; by means of the following two functions, the desired axis can then be select

[F3] + [↓]
[Axis+1] O
Select an axis for the display: **increment axis** within the active unit (only possible if the unit includes more than one axis)

[F3] + [↑]
[Axis−1] P
Select an axis for the display: **decrement axis** within the active unit (only possible if the unit includes more than one axis)

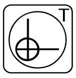
[F2] + [0]
[hand symbol] M
Change into the **programming mode** for nominal values;

abort with [ESC]

[F2] + [1]
Change into the **programming mode** for machine parameters;

abort with [ESC]; password: **9228**

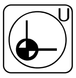
F1 + 1



Direct entry of a **reference value** for an axis to be selected (only possible in the reset state and if the machine parameter 3/**9** of that axis has been programmed accordingly, cf. section 4.6);

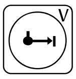
abort with ESC; the value becomes active with the next start signal

F1 + 2



Start a **reference search routine** for an axis to be selected (only possible in the reset state and if the machine parameters 3/**9** and 3/**11** of that axis have been programmed accordingly; additionally, High level must to be applied at /stop input P4/7/10; refer also to section 4.6.2)

F1 + 3



Direct entry of a **correction value** for an axis to be selected (only possible in the reset state and if the machine parameter 3/**7** of that axis has been programmed accordingly; refer also to section 4.7);

abort with ESC; the value becomes active with the next start signal

F1 + 4

Direct entry of the **park position** for an axis to be selected (only possible in the reset state and if the machine parameters 3/**56** and 3/**57** of that axis have been programmed accordingly; refer also to section 4.9);

abort with ESC; the value becomes active with the next start signal

F3 + +/−

Display the signal states (cf. section 4.5.7);

password: **9320**

F1 + 0

Error W

Display of stored fault message(s);

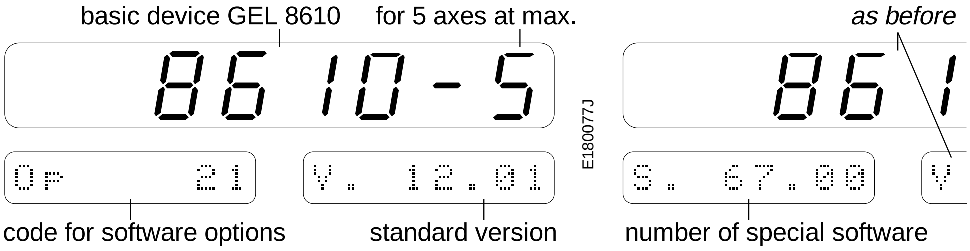
abort with ESC, scroll with the cursor keys, delete message with ; cf. section 6.2

[F3] + [·]

Display the actual software version (this is possible in all states of the automatic mode);

example:

a) with standard software b) with special software



<u>examples for software options</u> (cf. appendix C):

0p x1 : transmission protocol LB2
0p x2 : transmission protocols LB2 + GEL 131
0p 1x : circular interpolation
0p 2x : synchro control

[F1] + [·]
[clock key]

Display of date/time (only with serial interface – module V – with integrated real time clock);

abort with [ESC]; cf. section 4.12

[F2] + [⇨]
[card write key]

Write data to memory card (option): storing of operating and machine parameters (with security inquiry, refer also to storage location 1/**6** (GEL 8310) or 1/**9** (GEL 8610), respectively)

[F3] + [⇨]
[card read key]

Read data from memory card (option): loading of operating and machine parameters (only possible if the Controller is in the reset state for <u>all</u> units and the card has at least already once been written, with security inquiry)

[F1] + [⬇]
[hand key] + [>]

Manual operation: **slow speed forward** (only possible in the interrupted or reset state of the automatic mode and during teach-in operation; cf. storage locations 3/**18**...**24**);

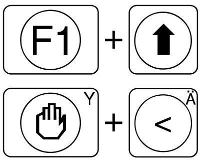
as this is an axis function, the axis to be controlled first has to be displayed (see further above: [F3]+[0] and [F3]+[⬇]/[⬆]);

High level must to be applied at the /stop input for the respective axis (P4/P7/P10)

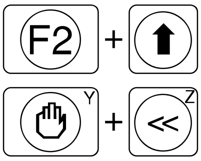
[F2] + [⬇]
[hand key] + [>>]

Manual operation: **fast speed forward**

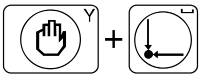
(see explanations of slow speed forward)



Manual operation: **slow speed reverse**
(see explanations of slow speed forward)



Manual operation: **fast speed reverse**
(see explanations of slow speed forward)



Manual preset of the signal ***actual = nominal*** (P27/P28/P29); as this is an axis function, the axis to be controlled first has to be displayed (see further above: F3 + 0 and F3 + ⬇/⬆)

3.1.2 Displays in the start and stop state

Display B	Display C	Description
3715.04 (display A)		
2.17.003	4500.00	standard display display B: axis.program.sentence display C: position/length nom. value
⬇ ⬆		
! pos.	4500.00	nominal position/length
!-= pos.	784.96	variance of position/length
! piece	50	nominal number of pieces [1]
= piece	16	actual number of pieces [1]
!-= pcs.	34	variance of number of pieces [1]
M. func.	110010	machine functions [1]
! speed	150.00	nominal speed rate [1]
Text	milling1	identification text for the selected axis (GEL 8310: input only via serial interface) [1] *under preparation*
Voltage	7.620	voltage at analogue output [2]
Delta_s	0.64	standard tolerance of position, contouring error [2]
! cycles	150	nominal cycles
= cycles	43	actual cycles
!-=cycl.	107	variance of the cycles
257.69	1586.27	act. values of 3 connected axes (here: display A = axis 2, display C = axis 3, display B = axis 1) [2]

E180062A

[1] These parameters can only be selected if the appropriate types of nominal values have been activated at the unit parameters, i.e., if they are part of a sentence.

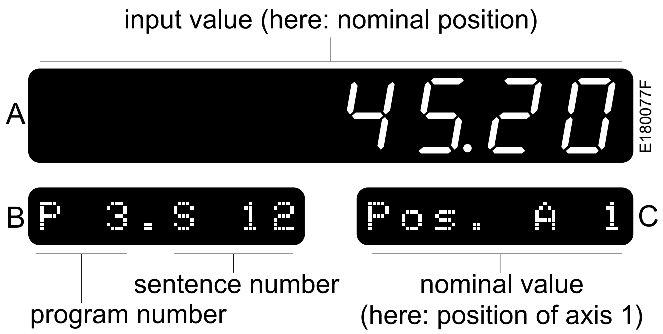
[2] These parameters can also be selected in the reset state.

3.2 Programming mode for nominal values

In this mode the position and control data required for the operation of the installation can be entered..

The programming mode can be protected against unauthorised data access by fixing a password (refer to appendix A, system parameters).

Display format:



Memory structure

The maximum of 7168 (GEL 8310) or 6416 (GEL 8610) storage locations for nominal values available can be devided at will up to 3 / 6 **units**, 99 **programs** per unit and 999 **sentences** per program.

Units are composed of up to 3 / 6 **axes** (to be determined via the system parameters, see appendix A) and have a common start, stop, and reset input each for the combined axes.

Each **program** can consist of a different number of sentences. When programming nominal values the end of program has to be defined after the last sentence via a special key combination (refer to the following section); it is handled within the system like an individual sentence which, however, only contains the number of program runs (cycles) to be set.

If the program shall be **extended** by some **sentences** at a later time, these must be inserted separately or as a group by copying (refer to the following section 3.2.1) to shift the non-replaceable end of program backwards. On the other hand, the end of program will be shifted forward if any sentences are deleted from the program.

The **structure** of the **sentence** is fixed separately for each unit (unit parameters, see appendix A). The following types of nominal values can be part of a sentence (in the given order):

- position/length for each of the combined axes (compulsory)
- number of pieces (batch counter)
- time for auto start
- machine functions
- speed rate for each of the combined axes
- identification text (for the GEL 8310 only in conjunction with a serial interface) – *under preparation*

In addition, a **path control** can be activated; see section 4.13.

The nominal value type **position/length** may be replaced by another parameter:

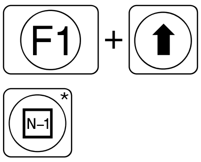
- program flow instruction or coordinates offset (cf. sections 4.14 and 4.15)
- a parameter depending on the inserted option (e.g. »radius« or »angle« with the Circular Interpolation option, refer to appendix O)

If the sentence or unit structure is to be changed at a later time, all programs of this unit will be deleted when you confirm the security inquiry `Delete unit ?` with [⇒] (*start* will cause no action then).

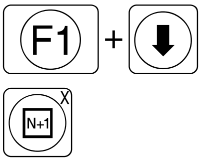
3.2.1 Functions

The following keys and key combinations are available in the programming mode for nominal values to call up certain functions:

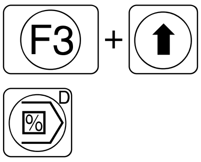
[F2] + [0] [Auto]	Terminate the programming mode of nominal values and change to **automatic mode**; if modifications have been made all program data are stored automatically (status message in displays B/C: `Saving program`); this also applies if the programming mode is left via [ESC])
[ESC]	**Cancel input** or **leave function** and return to the superior selection level; when creating a (new) program the security inquiry `Prog.end missed` will be issued (if you confirm with [⇒] all entries will be ignored; see [F3]+[9] further below)
[⬆], [⬇]	**Change up/down** to the next/previous nominal value (a value modified before will not be stored)



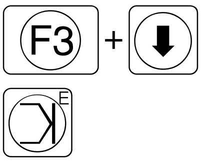
Jump to the **beginning** of the **previous sentence**



Jump to the **beginning** of the **next sentence**



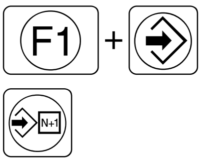
Jump to the **beginning** of **program**



Jump to the **end** of **program** (= number of program runs, cycles)

Confirm the **entry** made and change to the next nominal value

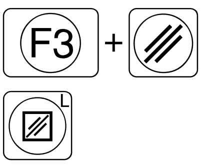
Respond to a security inquiry with "yes" (any other key means "no")



Confirm the **entry** made and change to the next nominal value of the **same type** within the program (quick input);

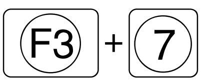
so you can input all the position values (e.g.) directly one after the other skipping the other nominal value types (for the moment)

Delete a preset or entered **value**



Delete a **sentence** (without security inquiry; only possible if the Controller is in the reset state for <u>all</u> units);

the numbers of the following sentences are decremented by 1



When selecting a program: **delete all programs** of the associated unit (with security inquiry; only possible if the Controller is in the reset state for <u>all</u> units)

F3 + 7 / [delete key F]

When selecting a sentence: **delete all sentences** of the associated program (with security inquiry; only possible if the Controller is in the reset state for <u>all</u> units)

F3 + [arrow key] / [insert key R]

Insert a sentence (only possible if the Controller is in the reset state for <u>all</u> units);

the numbers of the next sentences are incremented by 1; the new sentence is inserted after the actual one and contains as preset values the data of the actual sentence

F3 + 8 / [Copy key C]

Copy sentences within a unit (only possible if the Controller is in the reset state for <u>all</u> units);

the sentence range and the program are to be selected as source, the actual program is the destination; the sentences are copied either in replace or insert mode, depending on the completion of the sentence end number entry

– [arrow key] : overwrite the sentences following the actual one,
– F3 + [arrow key] / [insert key R] : insert the sentences before the actual one

In the first case, there must still be as many sentences after the actual sentence in the destination program as shall be copied; otherwise, an error message will be issued.

F3 + 0 / [key I], [key H]

Change between position and length: the entered position value is interpreted either as absolute position (absolute dimension processing, fixed zero) or as relative length (incremental dimension processing, continuous zero);

changing is only possible <u>before</u> the value entered is confirmed by pressing the [arrow key] key, and only with storage location 3/**44**=0 (system of absolute dimensions)

F3 + 0 / [key J]

Continuous sentence processing;

this function can only be activated for the nominal value type »speed rate« and is marked by an arrow (➡) in display C (refer to section 4.3)

F3 + 1

Select program flow instruction **CALL Pr.** (at the beginning of the sentence only: in place of the nominal value type »position« or »length«, see section 4.14.1)

F3 + 2 Select program flow instruction **JUMP Pr.** (at the beginning of the sentence only: in place of the nominal value type »position« or »length«, see section 4.14.2)

F3 + 3 Select program flow instruction **JMP sent** (at the beginning of the sentence only: in place of the nominal value type »position« or »length«, see section 4.14.2)

F3 + 4 Select program flow instruction **IF I/O** (at the beginning of the sentence only: in place of the nominal value type »position« or »length«, see section 4.14.3)

F1 + 1 **Absolute coordinates offset** (at the beginning of the sentence only: in place of the nominal value type »position« or »length«, see section 4.15)

F1 + 2 **Relative coordinates offset** (at the beginning of the sentence only: in place of the nominal value type »position« or »length«, see section 4.15)

F1 + 9 / **Teach In** (B) Activate **teach-in operation**;

this function can only be activated at nominal value type »position« (absolute dimensions) and is identified by an appropriate plain text in display C; the drive can be moved manually; when actuating the [→] key the actual value (display A) is taken over as nominal value and the teach-in operation is terminated; abort with ESC

F3 + 9 / **Def. Prog. End** (K) **Define** the **end of program**: fixing the number of program runs (cycles, 0 = unlimited);

this function can only be called up at the beginning of a sentence (at position/length); when confirming the value entered by pressing the [→] key, the programming mode is terminated

3.2.2 Programming example

The **position** of axis 1 in unit 1 (2 axes), program no. 3, sentence no. 12 is to be changed from 45.20 to **45.50**:

With the GEL 8610, key [M] can be used instead of [F2]+[0].

The password inquiry will only appear if an appropriate arrangement with respect to the machine parameters has been made (refer to appendix A, system parameters).

With the GEL 8610, key [Auto S] can be used instead of [F2]+[0].

3.3 Programming mode for machine parameters

In this mode the operating of the Controller, its adaptation to the plant and other properties are fixed. The appropriate machine parameters are enlisted and explained in a table in **appendix A**.

The machine parameters are hierarchically divided into 3 categories (planes):

- system parameters (plane **1**)
- unit parameters (plane **2**, subplanes 1...3 / 6)
- axis parameters (plane **3**, subplanes 1...3 / 6)

System parameters specify the basic operating mode of the Controller (the axes/unit configuration etc.).

Unit parameters specify for each unit consisting of one or more axes which nominal values shall be included in a sentence and which other preset values shall apply for all axes of this unit.

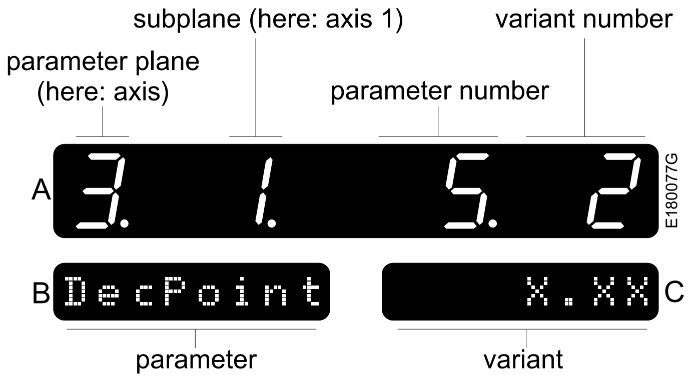
With the **axis parameters**, the Controller is matched to the different drives (axes): actual value processing, calibration, control, etc.

To avoid an unauthorised access to the data, the programming mode can only be activated after the entry of a certain figure code – the password **9 2 2 8**. It can be temporarily deactivated via storage location 1/**14** (GEL 8310) or 1/**17** (GEL 8610).

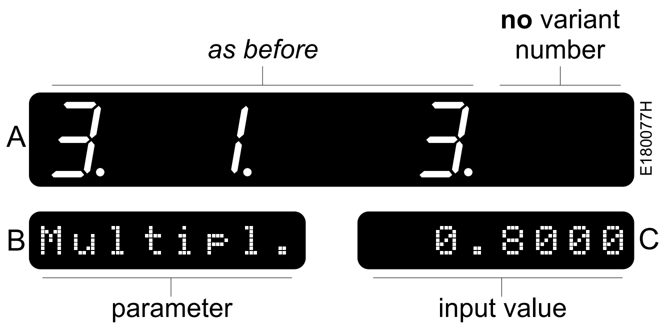
Once a storage location has been selected, the Controller switches into the **reset** state.

<u>Display format</u>:

a) Parameter with selection of variants



b) Parameters with value input



3.3.1 Functions

The following keys and key combinations are available in the programming mode for machine parameters to call up certain functions:

[F2] + [0]
[Auto]

Terminate the programming mode of machine parameters and return to **automatic mode**;

if modifications have been made the appropriate parameters will be actualised in the RAM of the Controller (status message in display B: Loader); this also applies if the programming mode is left via the [ESC] key

[ESC]

Cancel input or **abort selection function** and return to the next higher selection or programming plane

[⇨]

Confirm the **selection** made or **value entered** and change to the next parameter

Respond to a security inquiry with "yes" (any other key means "no")

[///]

Delete a stored or entered **value**

[⇧], [⇩]

Select the programming plane

Change up/down to the next/previous parameter within the associated programming plane (without storing a modified value or performed selection)

[F1], [F3]

Scroll the machine parameter **variants** forward/reverse (confirm with [⇨])

F3 + 7

Clear the **total data memory** (with security inquiry);

this is only possible directly after entering the programming mode, i.e., only inside of the top selection level;

stored faults are also deleted

F2 + 8

Copy

Copy axis parameters (only possible inside of the axis parameter plane);

at first, the axis whose parameters shall be used has to be selected (source axis);

the parameters of the active axis – whose parameters are just displayed – (destination axis) will be overwritten;

execute the copy function with [Enter key] or abort with [ESC]

F1 + ·

[Clock key]

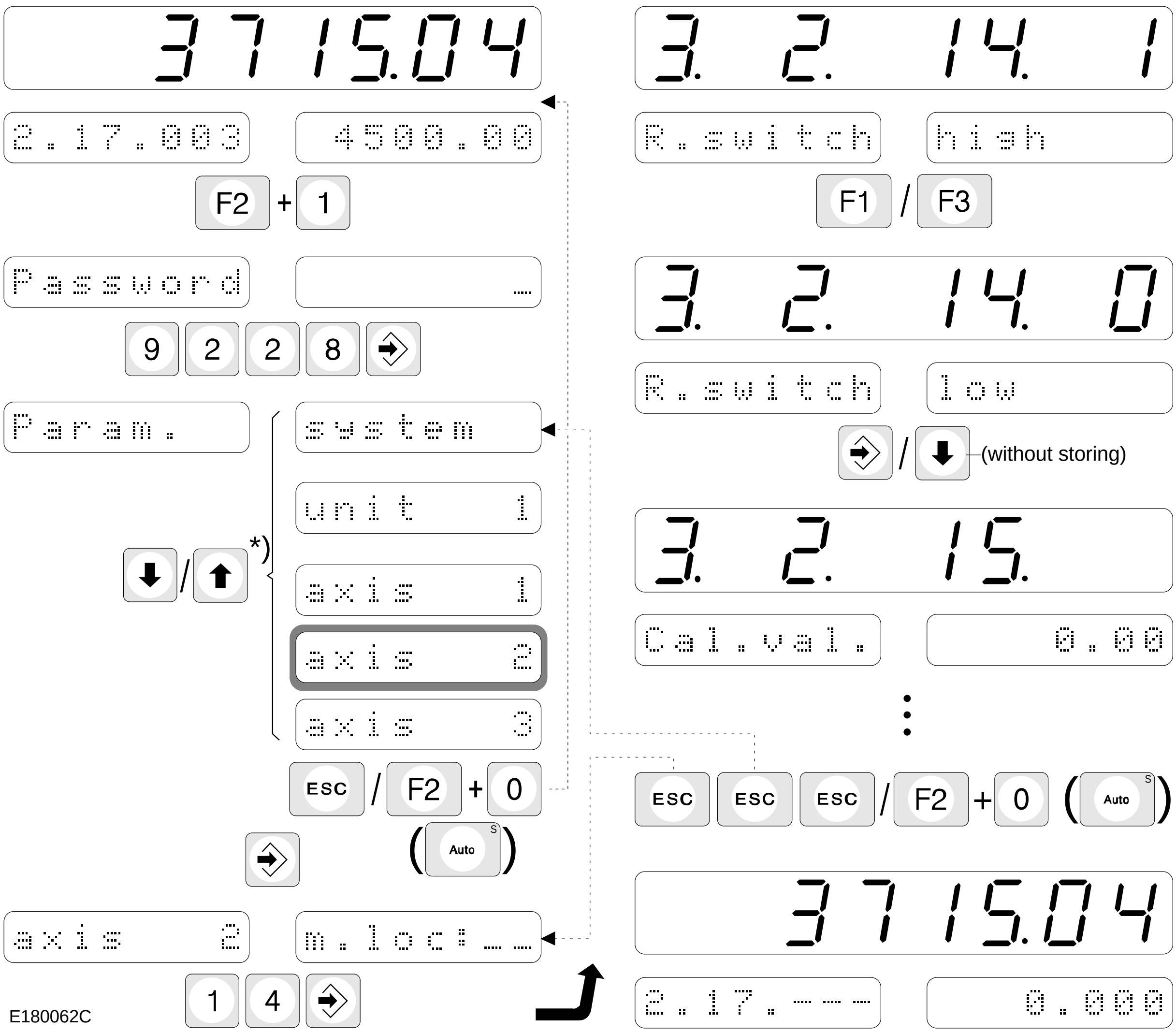
Set the date/time (with optional serial interface module with real-time clock only);

this is only possible directly after entering the programming mode, i. e., inside of the top selection level;

abort/terminate with [ESC], go on with [Enter key] (refer to section 4.12)

3.3.2 Programming example

The switching level of the *reversing switch* signal (storage location 3/**14**) for axis 2 shall be inverted (change of travel direction at Low signal):



---

*) It is assumed that 3 axes are connected and assigned to one unit (#1); therefore no other units or axes can be selected.

4 Functional Descriptions

4.1 Definitions

4.1.1 Actual unit of measurement

The actual unit of measurement is the unit of measurement which you use in the equipment: e.g. m, cm, mm, inch, degrees. With incremental encoders, it is determined by the **number of pulses** and the **edge evaluation** (storage location 3/**1**). In addition, for all encoders by the **multiplier** (storage location 3/**3**) and the programmed **decimal places** (= resolution, storage location 3/**5**). This also implies mechanical converters like measuring wheels or gearings.

The input of values is made like at a pocket calculator, i.e. for values in whole actual unit of measurement no point needs to be entered. The Controller fills missing decimal places automatically with zeroes.

Example for a resolution of 1 hundredth:

Value: 100 mm ⇒ entered: 100 or 100. or 100.0 or 100.00

⇒ always indicate: 100.00

Position values are always specified in actual unit of measurement. Nominal speeds can also be pre-defined either in actual unit of measurement per second or in an other unit (see section 4.2).

4.1.2 Display units (DispU)

In this manual, display units abbreviated to **DispU** mean a sequence of digits without decimal point as it is shown in the display A. With incremental encoders, this is the number of count increments after edge evaluation and multiplier. Example:

Value in display: 1234.00 ⇒ DispU: 123400 (actual unit of measurement: 1234)

4.1.3 Count direction

When specifying a direction (e.g. automatic reference positioning forward), the Controller and/or the regulation expects a determined behaviour of the actual value counter. Basically, the following assignment is valid:

forward ⇒ **incrementing** of the actual value, e.g.
… → -157 → … → -1 → 0 → 1 → … → 123 → …

reverse ⇒ **decrementing** of the actual value, e.g.
… → 123 → … → 1 → 0 → -1 → … → -157 → …

WARNING

On start-up:

If, with a regulated **forward positioning** (e.g. with reference positioning or positioning in the Controller-defined forward direction that can also equipment-specifically deviate)**,** the actual value counts **downward** then the contouring distance increases very quickly and thus the voltage at the analog output. This results in an **accelerated** and **almost no more controllable movement** in the **wrong direction**. Correspondingly, this is also valid for reverse positioning with upward counting.

On the supposition that the **position loop control** is **disabled** for the interrupted and reset state (storage location 3/**47** = **0**) then this process can be aborted either manually by specifying the *stop* or *reset* signal or automatically if a not too large maximum **contouring error** value has been programmed (storage locations 3/**42** and 3/**43**). Otherwise, only an **EMERGENCY STOP** will help!

Therefore, check whether the direction assignment i.e. the encoder/amplifier connection is correct. The measures to be performed for this are described in section 5.4.

4.2 Speed rates

In the machine parameters programming mode, speed rates are principally entered in **actual measuring units/sec** taking into account the resolution defined at storage location 3/**5**.

Example:

Actual measuring unit: **mm**, resolution: 1/100, speed: 45.5 **mm/sec**
⇒ input: ④⑤⊡⑤⊕, display: 45.50

The entry and display of **actual** speed rates in a **sentence** are possible in any length and time unit and with another resolution. This requires, however, to program 2 storage locations for the axis parameters accordingly: 3/**45** and 3/**46**.

The **multiplier** at storage location 3/**45** consists of two components:

a) physical conversion factor between the desired unit (e.g. cm/sec) and the standard unit (e.g. mm/sec): 1 cm/sec = 10 mm/sec

b) correction factor for the desired resolution of the speed values; the number of **decimal places** has to be fixed in storage location 3/**46**. The following facts apply:

- For each decimal place by which the speed resolution is **higher** than the actual measure resolution, the point of the value in a) must be shifted to the **left**, i.e. the value must be **divided** by 10.
- For each decimal place by which the speed resolution is **less** than the actual measure resolution, the point of the value in a) must be shifted to the **right,** i.e. the value must be **multiplied** by 10.

Example 1: In the equipment, the actual measuring unit **cm** is used with a resolution of 1/100 cm = 0.01 cm (storage location 3/**5**=2). The speed values shall be input in **m/min** with two decimal places:

storage location 3/**46** = storage location 3/**5** = 2 (»X.XX«)

storage location 3/**45** = (1 m/min) / (1 cm/sec) =
= (100 cm/60 sec) / (1 cm/sec) =
= 100 / 60 = 1,6667

If the speed is to be entered with only 1 decimal place then storage location 3/**45** = 1.6667 ∗ 10 = 16.6667. For 3 decimal places, storage location 3/**45** has to contain = 1.6667 / 10 = 0.1667.

Example 2: Irrespective of the actual measuring unit used in the equipment (e.g. mm, resolution 0.1 mm), the speed rates of the drive shall be entered in **revolutions/min** (**rpm**) without any decimal places and with 1 revolution corresponding to a distance of 10.0 mm:

storage location 3/**5** = 1 (»X.X«)

storage location 3/**46** = 0 (»X.«)

storage location 3/**45** = (1 revolution/min) / (1 mm/sec) ∗ 10 =
= (10 mm/60 sec) / (1 mm/sec) ∗ 10 =
= 10 / 60 ∗ 10= 1,6667

The above facts are represented again for those who prefer to work with mathematical formulas:

$$c_v = \frac{D}{A} * 10^{(DA-DD)} \qquad (c_v < 100)$$

c_V: conversion factor according to »Spd.mult« at storage location 3/**45**
D: desired unit of length or angle per each unit of time for the speed
A: actual measuring unit per second (default unit of the speed)
DA: number of decimal places in the actual value display (storage location 3/**5**)
DD: number of desired decimal places for the speed (storage location 3/**46**)

The following formula can be used for checking: $v_A = c_v * v_D$

v_A: speed in the Controller-internal format (actual measuring units/sec without decimal point: value in DispU)

v_D: speed in the desired format (value in DispU)

Example 1: In the equipment, the actual measuring unit cm is used with a resolution of 1/100 cm = 0.01 cm. The speed rates shall be entered in m/min with two decimal places:

$$D = m / min = 100 cm / 60 sec; \quad A = cm / sec; \quad DA = DD = 2$$

$$c_v = \frac{m / min}{cm / sec} * 10^{(2-2)} = \frac{100cm / 60 sec}{cm / sec} * 10^0 = \frac{100}{60} * 1 = \underline{\underline{1.6667}}$$

Check for e.g. $v_D = 60\, m / min\ (\hat{=}\ 6000\, DispU)$:

$$v_A = c_v * v_D = 1.6667 * 6000\, DispU = 10000\, DispU\ (\hat{=}\ 100.00\, cm / sec)$$

If the speed shall be entered with only one decimal place (DD=1) then $c_v = 1.6667 * 10^1 = \underline{\underline{16.6667}}$;

with 3 decimal places (DD=3): $c_v = 1.6667 * 10^{-1} = \underline{\underline{0.1667}}$.

Example 2: Irrespective of the actual measuring unit used in the equipment (e.g. mm, resolution 0.1 mm), the speed rates of the drive shall be entered in **revolutions/min** (**rpm**) without any decimal places. This corresponds to a distance of 10.0 mm with 1 revolution:

$$D = 1 / min\ \hat{=}\ 10mm / 60 sec; \quad A = mm / sec; \quad DA = 1; DD = 0$$

$$c_v = \frac{1 / min}{mm / sec} * 10^{(1-0)} = \frac{10mm / 60 sec}{mm / sec} * 10^1 = \frac{10}{60} * 10 = \underline{\underline{1.6667}}$$

Check for e.g. $v_D = 3000\ 1/min\ (\hat{=}\ 3000\, DispU)$:

$$v_A = c_v * v_D = 1.6667 * 3000\, DispU = 5000\, DispU\ (\hat{=}\ 500.0\, mm / sec)$$

Do not forget: Program the desired decimal places at storage location 3/**46**

4.3 Continuous sentence processing (positioning without stop)

Requirement: The nominal value type »speed rate« must be part of a sentence (storage location 2/**3** ≠ 0).

Activation: When entering speed values in the nominal values programming mode the [F3]+[0] key combination or the [I▶] key is to be pressed. Then an arrow ➡ is shown in display C. The function is switched off again by pressing [F3]+[0] or [I▶] once more.

For ‘normal’ positioning, the drive is accelerated according to a calculated characteristic, run at constant (working) speed and then decelerate to reach the exact position. For continuous sentence processing, a start signal is internally generated and the drive is accelerated or decelerated directly to the nominal speed defined in the next sentence. Two modes are available for this, depending on the programming of storage location 2/**3**:

- **Mode 1** (storage location 2/**3** = 1 or 3)
  The start signal is generated as soon as the Controller normally would enter the braking phase
- **Mode 2** (storage location 2/**3** = 2)
  The start signal is generated as soon as the control pre-set of the position is equal to the programmed nominal value. Thus, the machine is moved at a constant speed or accelerated to its nominal position, i.e., without braking phase

Variant 3 of storage location 2/**3** can only be used together with the linear path control (refer to below).

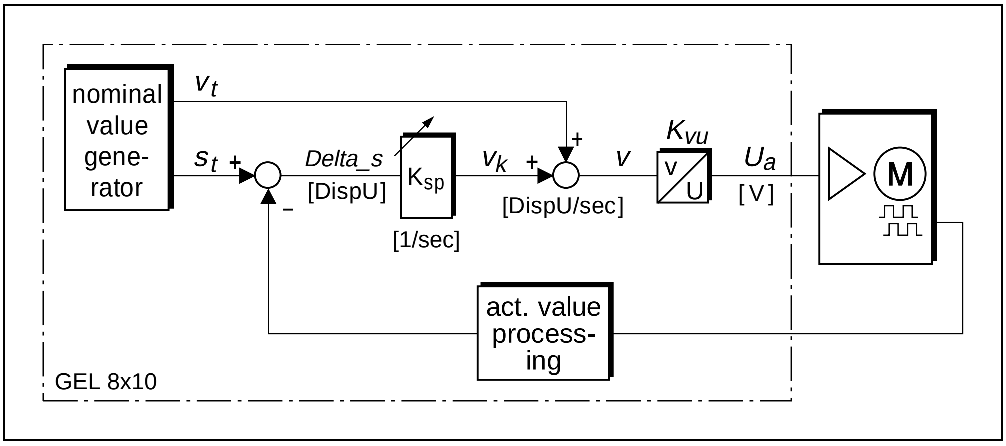
If several axes are combined to one unit (refer to storage locations 1/**3** to 1/**5** resp. 1/**8**), then a continuous sentence processing can be set for each axis. The drive that first fulfils the above-mentioned condition effects the generation of the internal start signal for all drives of the unit. These will then be directly accelerated or decelerated to their next nominal speed by considering the programmed jerk.

For active **path control** (refer to section 4.13), the following restriction applies:

> The programmed paths <u>must not show any break points</u> (⇒ use nearly continuous paths). Otherwise, a step voltage characteristic would be given for the drive or drives that have to run a shorter distance which they naturally cannot follow. That would result in an excessive mechanic or electric stress. With equal distances only the first drive in the unit will be accelerated or decelerated following the calculated characteristic, and the others are erratic.
>
> This restriction is not valid if variant 3 of storage location 2/**3** is programmed (spline function).

4.4 Drive control

4.4.1 Principle of regulation



E1800078

From the programmed machine parameters of the drive and the specified nominal values of position (and possibly speed), the Controller calculates a time-dependent speed characteristic $\boldsymbol{v_t}$ and the associated position $\boldsymbol{s_t}$.

At the beginning of each cycle (1 msec per programmed axis), the speed value $\boldsymbol{v_t}$ valid at that time is preset for the drive (principle of speed pre-control). At the same time, the actual position is inquired and compared to the value $\boldsymbol{s_t}$ calculated for this time. If the result shows a difference (positive/negative contouring error = Delta_s, to be shown in display C), then this difference is converted using the proportional action factor $\boldsymbol{K_{sp}}$ to an equivalent speed value $\boldsymbol{v_k}$. This value is then added to the time-dependent nominal speed value $\boldsymbol{v_t}$. The control dynamics (K_{sp}) can be adjusted according to requirements (storage location 3/**33**).

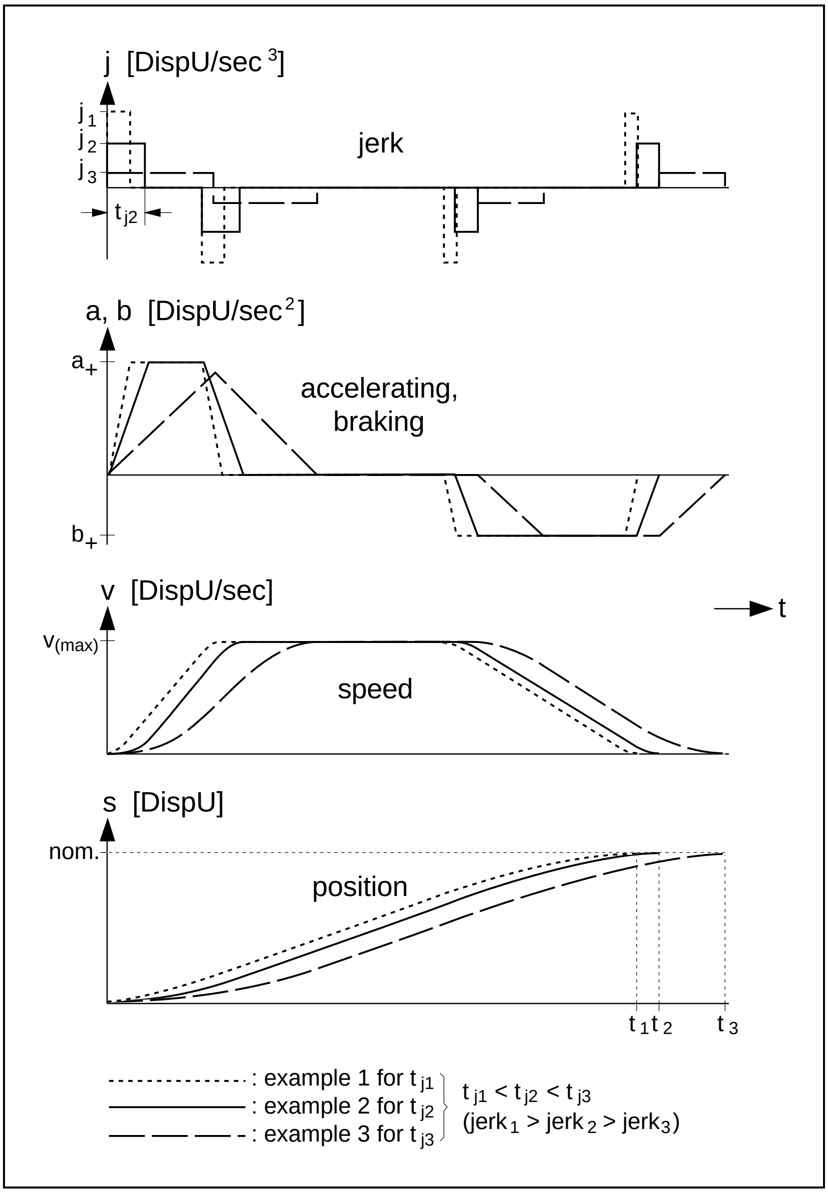
The ratio of the speed v to the output voltage U_a ($\boldsymbol{K_{VU}}$) is fixed by the programmed values of U_{min+}, U_{max} and v_{max} (storage locations 3/**29**, 3/**31** and 3/**32**).

After the control preset value has reached the value for the nominal position (speed pre-control is terminated then), the **dead range** (S_{dead+}/S_{dead-}) preset via storage locations 3/**27** and 3/**28** becomes effective: as long as Delta_s is <u>within</u> this range, U_a = 0 remains unchanged. This also applies to active position loop controlling in the interrupted or reset state (storage location 3/**47**).

4.4.2 Positioning characteristic

The characteristic of the positioning curve (e.g. soft start) is determined by the **jerk** parameter, i.e., the jerking time (refer to storage location 3/**39**).

The larger the jerking time is programmed, the smaller is the jerk, i.e., the softer the drive runs during starting and stopping. The following diagram illustrates these relations by the example of 3 different jerking times (t_j):



E180016B

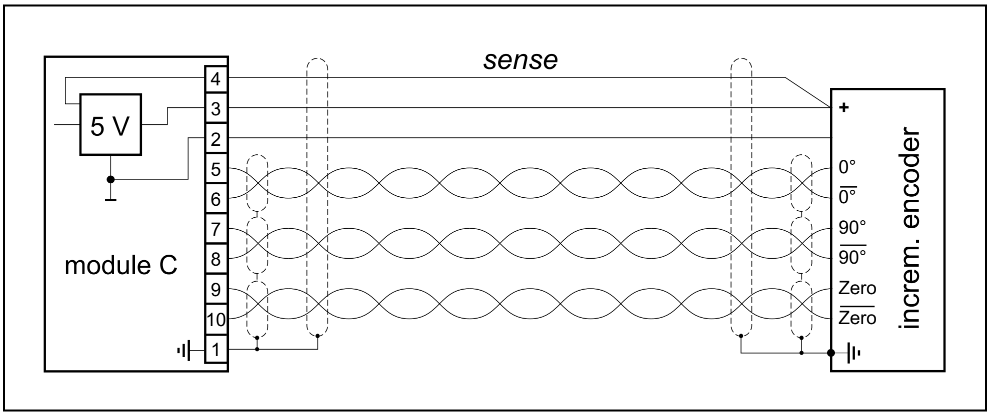
4.5 Signals

4.5.1 'Sense'

The sense line (module **C**) is a measuring line which measures the voltage drop across the positive line to the incremental encoder. Precondition is that both supply lines have an equal cross section and an equal length resulting in an the equal voltage drop. The output voltage of the Controller is then adjusted so that 5 V are applied to the encoder terminals.

If the sense input is not wired then the regulator supplies a stabilized output voltage of 5 V.

The following figure shows the possible connection of a 5 V encoder to the first count input of module C:



E180014C

4.5.2 'Fault'

The */fault* signal at terminal 30 of the (1st) module **P**1 becomes active (level changes from High to Low) when the following situations occur:

- the »Delta_s« contouring distance exceeds one of the programmed $S_{max+/-}$ values (storage locations 3/**42** and 3/**43**)
- a limit switch triggers (refer to section 4.11)

At the same time

– the Controller changes into the stop or reset state,

– the separation points in display B are flashing (only visible if the corresponding axis is just displayed).

This state remains until either the *start* or *search for reference* signal is applied or the drive will be manually positioned.

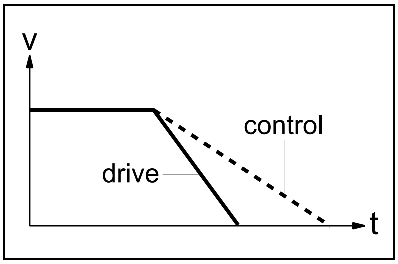
4.5.3 ‘Zero Delta_s’

Using the *zero Delta_s* signal at terminal 23 of the (1st) module **P**1, a contouring error can be reset that has been built up in the interrupted or reset state.

The function has been created for the case that

- the position loop control is activated for the interrupted state (storage location 3/**47** = »active«) and
- EMERGENCY STOP circuits are used that influence the drive system but set the Controller only into the interrupted state.

In a triggered condition, the drive does not follow (timely-limited) the regulation presettings of the Controller for decelerating anymore.



E180018A

This results in an increased contouring distance (Delta_s) causing a correspondingly high control voltage.

No fault is generated in the Controller if Delta_s stays below the S_{max} maximum value (storage location 3/**42/44**). When powering-up the drive system the next time, the control voltage generated then by the Controller may cause an inadmissible high loading of the drive.

To prevent this, the contouring distance for the corresponding axis can be set to zero by applying High level to terminal $P_1$23.

Precondition for the effectiveness of the signal:

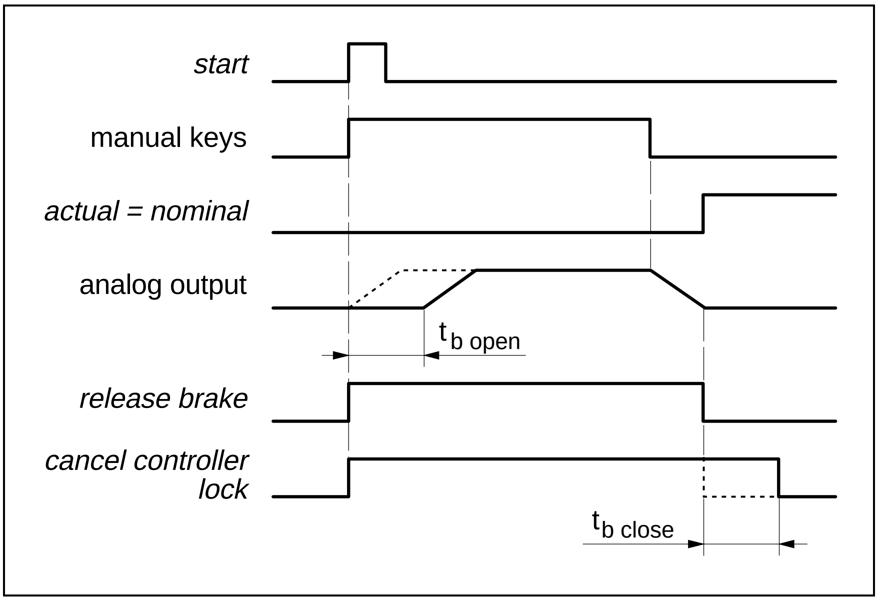
- parameter 3/**90** of the corresponding axis is/axes are »active«
- unit of the corresponding axis (axes) is in the interrupted or reset state

4.5.4 Drive control signals

The drive control signals include

- *release brake*
- *cancel controller lock* (*drive enable*)

The signals are output at terminal strip **D**. The following diagram shows the correlation:



The values for $t_{b\ open}$ (time to open the brake) and $t_{b\ close}$ (time to close the brake) are programmed at storage locations 3/**51** and 3/**52**.

4.5.5 Program processing signals

The program processing signals include

- *sentence end*
- *block end*
- *program end*
- *stop*
- *(reset)*

These signals are unit-related. They occupy a certain BCD decade (4 bits) at a data output module (A, L, or R) which has to be determined using storage location 2/**9** (refer to the Pin Layout in appendix B).

The signal *reset* can only be output in lieu of the signal *sentence end* or *block end*.

Sentence end is output with a start signal if the actual number of pieces is equal to the nominal number of pieces. If the number of pieces is not part of a nominal value sentence (storage location 2/**1** ≠ 1 or 3) then the signal is output at the start of each sentence, i.e., the signal remains always set.

Block end is output with a start signal if the actual number of pieces of the last sentence within a program run (cycle) has reached its nominal value. If the number of pieces is not part of a nominal value sentence (storage location 2/**1** ≠ 1 or 3) then the signal is output at the start of the last sentence.

Program end is output with a start signal if the actual number of pieces of the last sentence in the last program run (cycle) has reached its nominal value. If the number of pieces is not part of a nominal value sentence (storage location 2/**1** ≠ 1 or 3) then the signal is output at the start of the last sentence in the last program run.

All 3 signals remain also set in the interrupted (stop) state but not in the reset state. Alternately, the signal output can also be performed with the *actual = nominal* signal (refer to storage location 2/**11** in appendix A).

Stop is output if the Controller is in the interrupted or reset operating state. As long as the drive is manually positioned or is in the automatic reference search routine, the signal is reset for the appropriate unit.

Reset is only available if the »Sent.end« (1) or »BlockEnd« (2) variant has been programmed on storage location 2/**12**. The signal is output if the Controller is in the reset operating state. It is reset for the corresponding unit as long as the drive is manually positioned or is in the automatic reference search routine.

4.5.6 Range signals

These signals are axis-related. They occupy a certain BCD decade (4 bits) at a data output module (A, L, or R) which has to be determined using storage location 3/**61** (refer to the Pin Layout in appendix B).

For each of the 4 ranges R1…R4, one start value and one end value must be programmed (storage locations 3/**63**...**70**).

The signals can be assigned certain features regarding the interpretation of the programmed values and the function of signals (storage location 3/**62**):

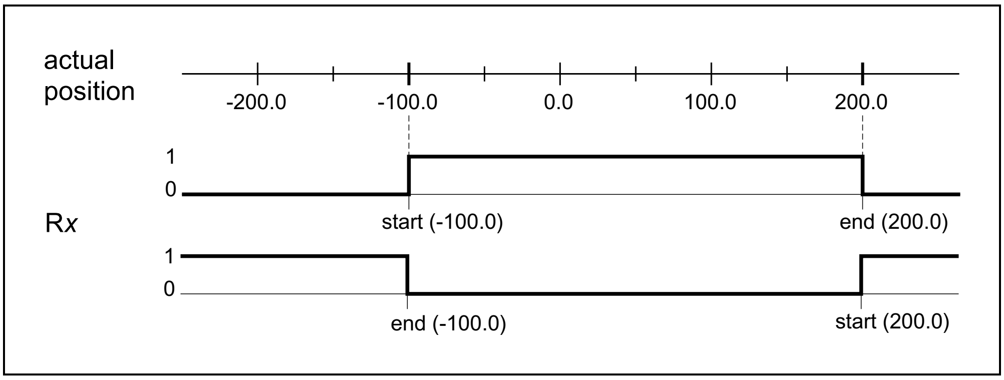
- values for the start and end of the range are absolute positions
- values for the start and end of the range are relative positions which are related to the difference ‘nominal – actual’ (variance)
- as before but signals are configured for controlling fast/slow-speed drives

The following distance and time diagrams show the relation as examples whereas R*x* refers to one of the R1 to R4 range signals and ‘start’ and ‘end’ characterize the values programmed for this.

Absolute ranges

Storage location 3/**62** = 0

Signals are output in all operating states.

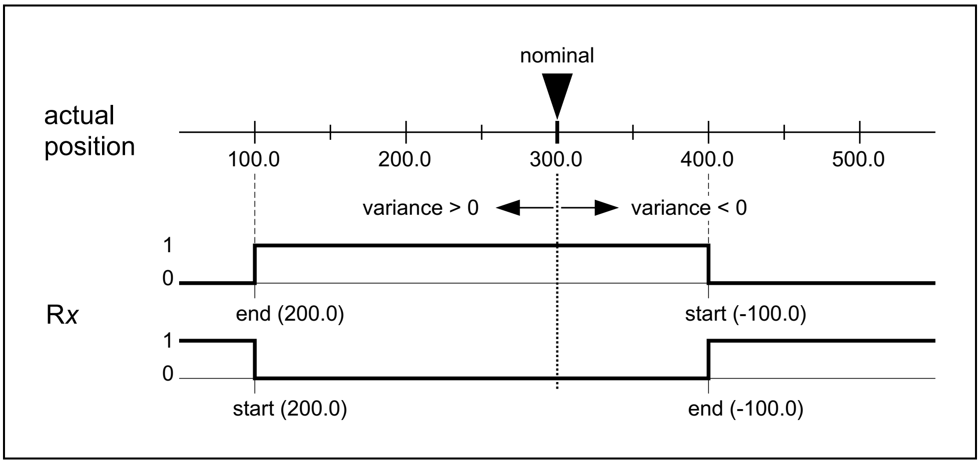


E180010A

Relative ranges

Storage location 3/**62** = 1

Signals are output in the started state only.



E180010B

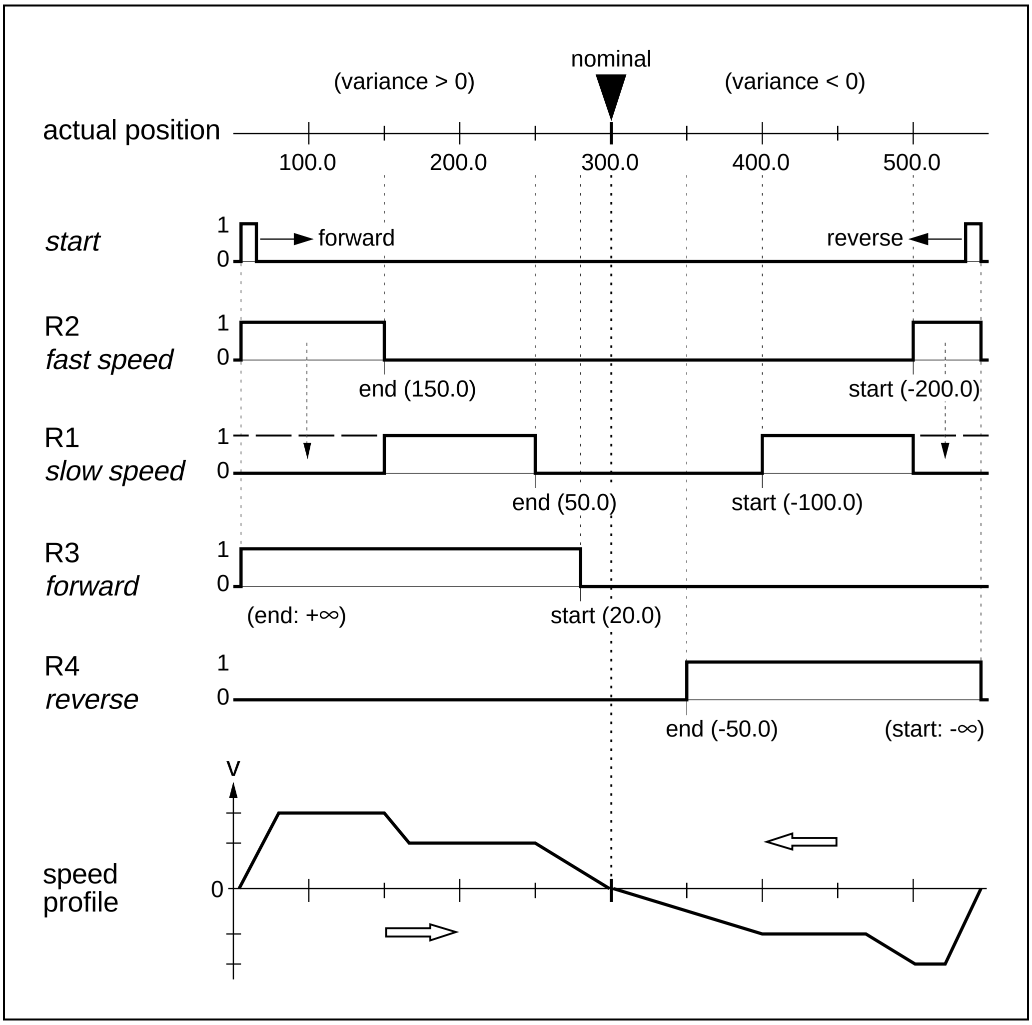
Drive signals

Storage location 3/**62** = 2

The range signals are used to control fast/slow-speed drives and have a fixed meaning (refer to the following diagrams). The values for the start and end are relative and related to the nominal position.

a) <u>Positioning</u>

The end value of R3 as well as the start value of R4 are set internally to a maximum value. Programmed values are ignored.



E180010C

<u>Storage location assignment</u>:

slow speed	3/**63** (R1:Beg.) = -100.0	3/**64** (R1:End) = 50.0
fast speed	3/**65** (R2:Beg.) = -200.0	3/**66** (R2:End) = 150.0
forward	3/**67** (R3:Beg.) = 20.0	3/**68** (R3:End) = —
reverse	3/**69** (R4:Beg.) = —	3/**70** (R4:End) = -50.0

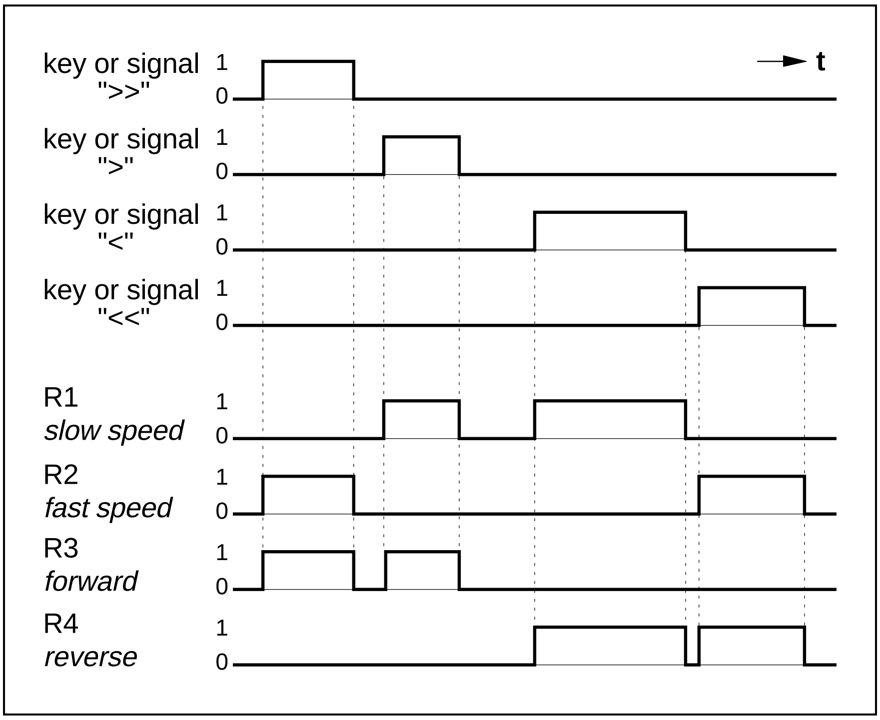
b) <u>Manual positioning of the drive</u>

Here, the drive signals are directly set with the appropriate control keys or signals.

Programmed start and end values are ignored.

The manual drive control is only possible in the interrupted or reset state of the automatic mode and during the teach-in operation when programming nominal values.

The following diagram is based on the fact that the polarity for manual controlling has not been changed (storage location 3/**20** = 0).



E180010E

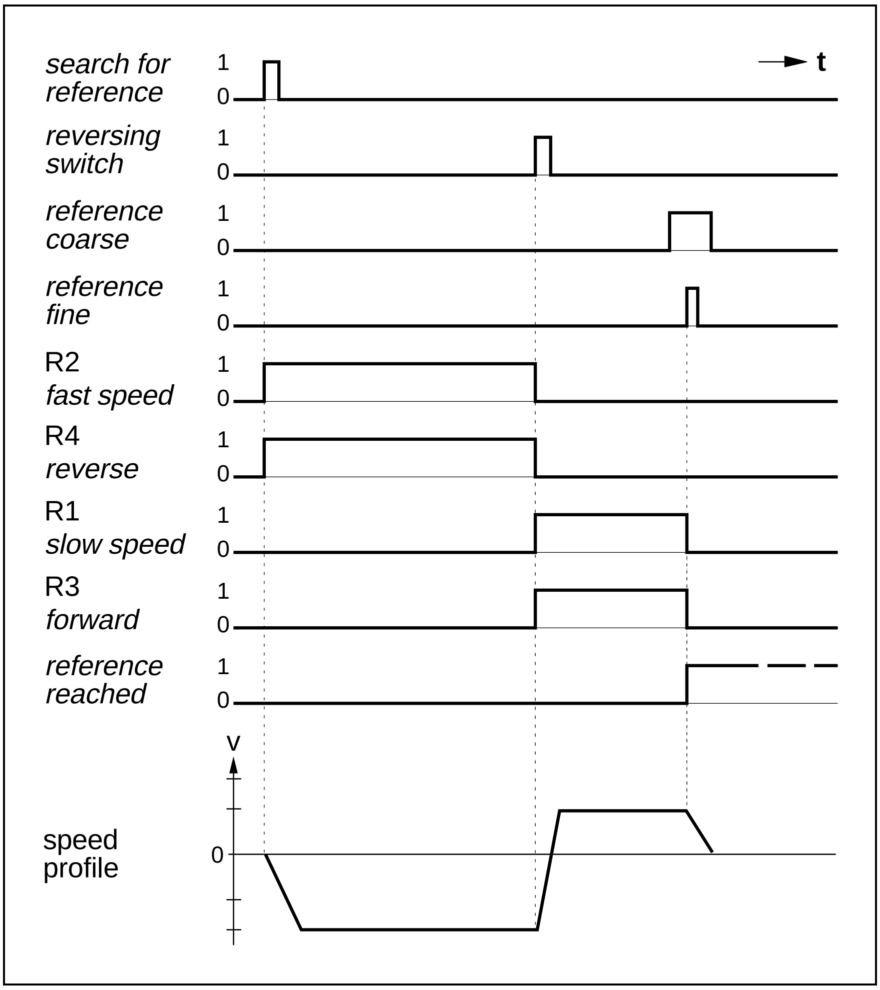
c) Automatic reference search routine

Here, the drive signals are set or reset directly via the appropriate control signals like *search for reference*, *reversing switch* and *reference fine*.

Programmed values for the start and end are ignored.

Further explanations on reference search routine are contained in section 4.6.2.

The following diagram assumes that the reference measure is set at forward traversing (storage location 3/**11** = 1).



E180010D

4.5.7 Display of the signal status

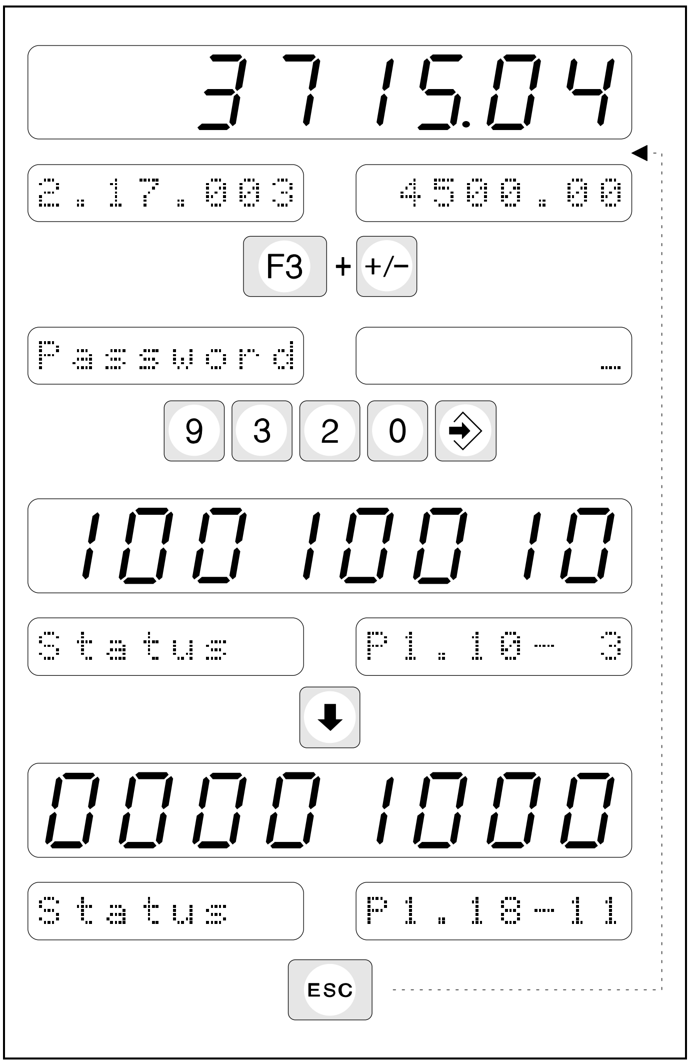
If you press the key combination [F3]+[+/−] and then enter the password '**9320**' you can display the signal states at the modules in groups of up to 8 signals. This is possible for each operating state of the automatic mode.

The displays show the following information:

- display A: max. 8 signal states, 0 = Low, 1 = High
- display B: text Status
- display C: identification of the module and the connections/signals/decades

Via the keys [↓]/[↑] you can scroll through the various display groups; [ESC] will abort the function.

<u>Example</u>: show the signal states at terminals 11 ... 18 of module P1



E180062D

High level at:

- P4 (/*stop* unit 1)
- P7 (/*stop* unit 2)
- P10 (/*stop* unit 3)
- P14 (*reference coarse* axis 1)

<u>Listing of the representable signals</u>:

Display C	Description

control inputs/outputs (8 or 6 digits)

P1.10- 3 P1.18-11 P1.24-19 P1.30-25	module P_1, terminal 10-9-8-7-6-5-4-3 " " 18-17-16-15-14-13-12-11 " " 24-23-22-21-20-19 " " 30-29-28-27-26-25
P2.10- 3 P2.18-11 P2.24-19 P2.30-25	**GEL 8610 only**: module P_2, terminals as above

drive control (4 digits)

D1.10- 7 D1.20-17 D1.30-27	module D_1, terminals 10-9-8-7 " " 20-19-18-17 " " 30-29-28-27
D2.10- 7 D2.20-17 D2.30-27	**GEL 8610 only**: module D_2, terminals as above

signals *reference fine* (3 digits)

Z1.RF3-1	module Z_1/C_1, terminal 29-19-9 (axis 3, 2, 1)
Z2.RF6-4	**GEL 8610 only**: module Z_2/C_2, terminal 29-19-9 (axis 6, 5, 4)

data output (8 digits)

A1.D1/D0 A1.D3/D2 A1.D5/D4	module A_1, pin 17-4-16-3-15-2-14-1 (decades 1, 0) " " 21-8-20-7-19-6-18-5 (decades 3, 2) " " 25-12-24-11-23-10-22-9 (decades 5, 4)
A2.D1/D0 A2.D3/D2 A2.D5/D4	module A_2, pins as above
A3.D1/D0 A3.D3/D2 A3.D5/D4	module A_3, pins as above
A4.D1/D0 A4.D3/D2 A4.D5/D4	**GEL 8610 only**: module A_4, pins as above

→

Display C	Description
data input (8 digits)	
E1.D1/D0 E1.D3/D2 E1.D5/D4	module E1, pin 17-4-16-3-15-2-14-1 (decades 1, 0) " " 21-8-20-7-19-6-18-5 (decades 3, 2) " " 25-12-24-11-23-10-22-9 (decades 5, 4)
E2.D1/D0 E2.D3/D2 E2.D5/D4	module E2, pins as above
E3.D1/D0 E3.D3/D2 E3.D5/D4	module E3, pins as above
E4.D1/D0 E4.D3/D2 E4.D5/D4	**GEL 8610 only**: module E4, pins as above

4.6 Reference measure

When using incremental encoders, a reference position to which all other positions to be positioned to are related to can/must be determined within the working range of the machine.

Concerning the power failure security (storage location 1/**2**) it can be determined whether incremental axes are to be calibrated first after powering-up. It is possible to exclude individual axes from this calibration (refer to storage location 3/**91**).

When setting the reference measure, one of two possible values is loaded into the actual value counter (display A). The signal state at terminal **$P_1$22** determines the active reference value (***reference2/1*** signal):

- **Low** level (or not connected): **1st** reference measure is active, value from storage location 3/**15**
- **High** level: **2nd** reference measure is active, value from storage location 3/**85**

The reference measures can either be programmed in the programming mode of machine parameters or directly in the reset state of the automatic mode via the [F1]+[1] key combination (precondition: storage location 3/**9** = 1 [»ref.val.«] or = 3 [»val/auto«]).

<u>Direct entry of a reference measure</u>:

- Press the [F1]+[1] key combination resp. the [⊕] key (GEL 8610)
- Enter the axis number and confirm with [⇒]
- Enter the reference value (display A) and confirm with [⇒]

  The input value is stored either into storage location 3/**15** or 3/**85** depending on the signal state at terminal **$P_1$22** (*reference2/1*, refer to above).

There are two possibilities of setting the reference measure. The selection is made via storage locations 3/**10** and 3/**11** (refer to below):

- when positioning the drive (3/**10**)
- with an automatic reference search routine (3/**11**)

4.6.1 Setting of the reference measure when positioning the drive

If the value programmed at storage location 3/**10** is unequal to 0, the reference measure can be set during the positioning process or the manual positioning of the machine. The storage location determines the **direction of travel** where the reference measure is to be set: for the forward motion (»forward«) or the reverse motion (»reverse«) or irrespective of the direction (»forw/rev«).

The reference measure is set with the positive or negative **edge** of the ***reference fine*** signal at terminal strip C or Z (proximity switch or zero signal of the encoder) if at the time of the edge the ***reference coarse*** signal is active. The edge resp. the level of the signals are determined at storage locations 3/**12** and 3/**13**.

By setting the reference measure, the Controller outputs the signal ***reference reached***. This signal is **reset**

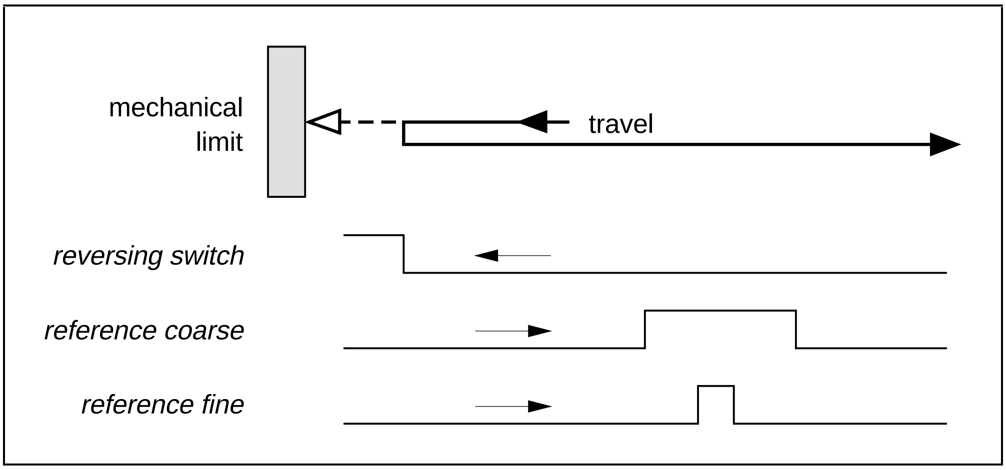
– at power failure,
– after presetting of the signals
  • *start* and
  • *reset* or
– after selecting a storage location in the machine parameters programming mode.

4.6.2 Automatic reference search routine

If a value unequal to 0 has been programmed at storage location 3/**11** and High level is applied to the *stop* input of the respective unit, a reference search routine for the machine can be initiated in the reset state of the automatic mode. The storage location determines the **direction of travel** where the reference measure is to be set: when moving forward (»autoforw«) or reverse (»auto rev«).

There are two alternatives to trigger the automatic reference search routine:

- by a positive signal **edge** at the ***search for reference*** input for the desired axis
- via the keyboard. In this case, the storage location 3/**9** must be programmed with 2 (»auto cal«) or 3 (»val/auto«). Start with [F1]+[2] resp. [key] (GEL 8610) and enter the axis number. If **0** is entered then all axes programmed as mentioned above (3/**9**) are simultaneously started for the reference search routine.



E1750038

After initiating, the machine is driven in such a way that it travels contrary to the direction determined at storage location 3/**11**. The corresponding speed rate is set at storage location 3/**16**.

When the reversing switch is reached (*reversing switch* signal becomes active, logic level determined at storage location 3/**14**), the direction of travel is reversed and the speed rate is reduced to the value determined at storage location 3/**17**.

The drive now moves towards the reference position. Once this point has been passed (*reference coarse/fine* signals), the actual value counter (display A) is set to the reference measure, the drive is stopped and the ***reference reached*** signal is output. This process is described in the previous section 4.6.1.

During the automatic reference search routine display B shows Ref instead of the sentence number.

The automatic reference search routine can be **aborted** by a */stop* or *reset* signal.

4.7 Correction value

To compensate cutting losses or tool wear, a positive or negative value can be preset by which each nominal position or nominal length will be corrected taking the sign into account. This applies to the system of absolute dimensions (storage location 3/**44**=0) as well as the system of incremental dimensions (storage location 3/**44** ≠ 0):

a) absolute dimensions, positions

target position = programmed nominal position + correction value

b) absolute dimensions, lengths (floating zero processing)

target position = previous nominal position/length + programmed length + correction value

c) incremental dimensions

counter value at *start* set to: - correction value (- residual value),

target position = programmed nominal length

The correction value is programmed at storage location 3/**6**, either in the machine parameters programming mode or directly in the reset state of the automatic mode (precondition: storage location 3/**7** = 1 [»active«]).

Direct entry of a correction value:

- Press the [F1]+[3] key combination resp. the [⟷] key (GEL 8610)
- Enter the axis number and confirm with [⇨]
- Enter the correction value (display A) and confirm with [⇨]

  The input value becomes active with the next start signal.

It is also possible to preset the correction value via an optional data input module (refer to storage location 3/**76** in appendix A and section 4.10.1).

4.8 Rotary table positioning

The rotary table positioning is activated by programming a value at storage location 3/**8**.

The special feature of this type of positioning is, apart from the restricted counting range for incremental encoders, the **path optimization**. Here, the Controller itself selects the travel direction depending on the actual position and the next nominal position. Thus, always the shorter distance is travelled.

The rotary table function does not work together with the

- path control (storage location 2/**5**),
- system of incremental dimensions (storage location 3/**44**).

A parameter error message (`Param. error`) is displayed if one of these functions has been programmed.

For the **input monitoring** of the nominal positions/lengths, the minimum value is internally set to zero and the maximum value to 'rotary table range - 1 DispU' independent of the values programmed at storage locations 3/**71** and 3/**72**. In addition, the software limit switch function will be ignored if it is activated (storage location 3/**73**).

Counting example for a defined rotary table range of 360.0 actual measuring units:

counting up: … → 359.8 → 359.9 → 0.0 → 0.1 → 0.2 → …
counting down: … → 0.2 → 0.1 → 0.0 → 359.9 → 359.8 → …

The value to be **programmed** and the **operating** value of the rotary table range may differ. This is caused by taking a multiplier value at storage location 3/**3** into account. In addition, a four-fold edge evaluation is always taken as basis independent of the one programmed at storage location 3/**1** (applies only to incremental encoders). Thus, a possible shift of the zero crossing is avoided which can occur due to the internal binary processing of a fractional multiplier for the incoming encoder pulses or absolute positions.

$$\text{Programmed range (3/8)} = \frac{\text{Operating counting range}}{\text{Multiplier}} * \frac{4}{\text{Edge evaluation}}$$

Example of an incremental encoder with 10.000 pulses/revolution:

count. range of rotary table in actual measuring units:... 360.0
edge evaluation (3/**1**): .... single (factor 1)
mechanical transmission: 1:1
multiplier (3/**3**): 0.3600

value to be programmed (3/**8**) = (360.0 / 0.3600) ∗ (4 / 1) = 4000.0

Only positive values can be entered for the setting of lengths nominal values. In operation, negative lengths are, however, achieved for values that are larger than half of the counting range. Thus, a certain travelling direction can be forced for the drive by an appropriate selection of the nominal length.

Example with the data from the previous example:

desired nominal length: -100.0 (the drive shall move backward by 100.0)

nominal value to be entered: 360.0 – 100.0 = 260.0 (> 360.0 / 2 !)

4.9 Parking

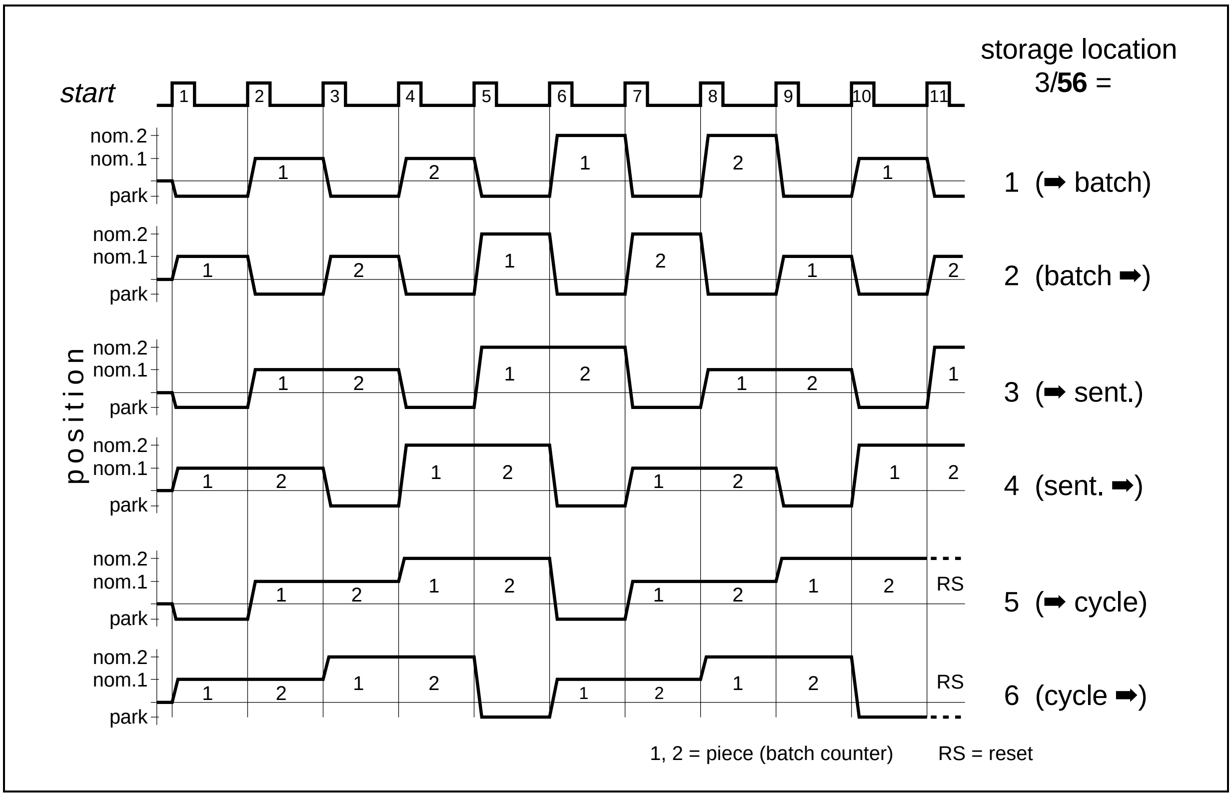
The parking position is an additional nominal position. Its **value** is to be programmed at storage location 3/**58**, either in the machine parameters programming mode or directly in the reset state of the automatic mode (precondition: storage location 3/**57** = 1 [»active«] and 3/**56** ≠ 0 [»inactive«]). Direct entry via F1+4 and determination of the axis number.

The parking position is only taken into account if the Controller operates in the system of absolute dimensions (storage location 3/**44** = 0).

The **speed rate** for reaching the parking position is to be determined at storage location 3/**59**.

While travelling to the park position and remaining there, display B shows Par instead of the sentence number. At the same time the machine functions programmed at storage location 3/**60** are output at the fixed data output module (refer to appendix A).

Storage location 3/**56** specifies when the parking position is to be reached. The following diagram gives an example illustrating the possibilities for a program with two runs (cycles) consisting of two sentences (positions) with the nominal piece number of 2:



E180040A

4.10 External data input/output (option)

The data modules used must be addressed correctly, so that data will be available at the desired module. Address coding is performed through particular jumpers on the boards; in appendix B you will find the necessary information.

Note: In appendix Y you will find a form, where you can put down the actual configuration of your data inputs and outputs.

4.10.1 Data input

For presetting data (BCD) and signals, up to 4 modules **E** are available, each with 24 logic inputs grouped into 6 decades with 4 bits each and enumerated from 0 to 5 (cf. appendix B).

The input data can be unit- or axis-related. Accordingly, programming is done either with the unit parameters (**2**/x) or the axis parameters (**3**/x). The following possibilities exist:

- program number (storage location 2/**6**)
- sentence and program number (storage location 2/**7**)
- manual drive signals (storage location 3/**19**)
- hardware limit switches (storage location 3/**74**)
- nominal position/length (storage location 3/**75**)
- correction value (storage location 3/**76**)
- speed rate (storage location 3/**77**)

With the next start signal, the **BCD data** (nominal values, program and sentence number) are taken over by the Controller. Nominal values are only effective if a program has already been selected.

All nominal values must be specified in **DispU**, i.e., the resolution used is not taken into account (e.g., for a speed rate value of 150 with the resolution of 1/100, i.e. 150.00, the BCD value 15000 has to be input). You will find further explanations at the individual storage location descriptions in appendix A.

If identical input positions (decades) are programmed for different units or axes then the same data will be interpreted differently. Therefore, you must plan and program the assignment of units, axes and data inputs/types very carefully.

Combinations of data groups are possible. This allows a more effective use of a data module. For instance, with the corresponding programming, the program numbers for three units or the signals for the manual positioning of six axes may be preset at a single data input connector (you will find a more detailed example performed for the data output at the end of the next section).

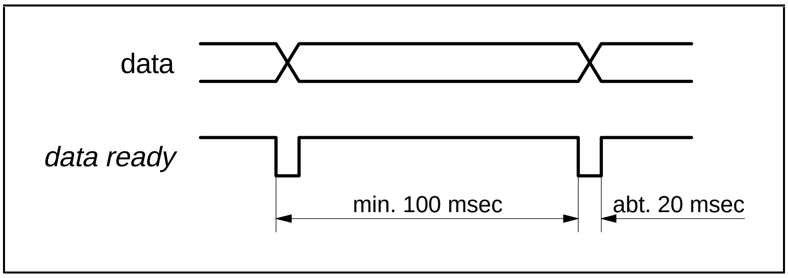
4.10.2 Data output

There are up to 4 different modules available for the output of (BCD) data and signals:

- 24 logic outputs (plug **A**)
- 24 power outputs (terminal strip **L**)
- 12 relay contacts (terminal strip **R**)

The logic and power outputs are grouped into 6 decades with 4 bits each and enumerated from 0 to 5 (cf. appendix B). The first three decades are also output at the relay contacts.

The output of **nominal/actual** values (position, correction value) may include the ***data ready*** signal at the most significant bit (MSB) position if programmed accordingly (refer to storage location 3/**80** and following). Low level will be output as long as the data are not stable:



E180033C

In addition, a **sign** bit can be specified for the output. If not specified, the full 6 decades are available for the value. In the other case, the sign is output as MSB or when evaluating the *data ready* signal as MSB–1. The value ranges are reduced accordingly. A **minus** sign is assigned **High** level.

The output data can be unit- or axis-related. Accordingly, programming is performed either at the unit parameters (**2**/x) or at the axis parameters (**3**/x) (refer to below).

If identical output positions (decades) are programmed for different units or axes all data are **logically OR**ed (bit-by-bit) before they are output.

The following data and signals can be output:

- machine functions (storage location 2/**2**)
- program and sentence number (storage location 2/**8**)

- program processing signals (storage location 2/**9**)
- range signals (storage location 3/**61**)
- actual position (storage location 3/**81**)
- nominal position (storage location 3/**80**)
- correction value (storage location 3/**82**)

With an appropriate selection of the decades, you can output different types of data at one single data output. This allows a more effective use.

Example: Four machine functions, program and sentence number (the latter with two digits, at maximum: 99) and the program processing signals of **one** unit shall simultaneously be available at the first data output **A1**.

Programming:

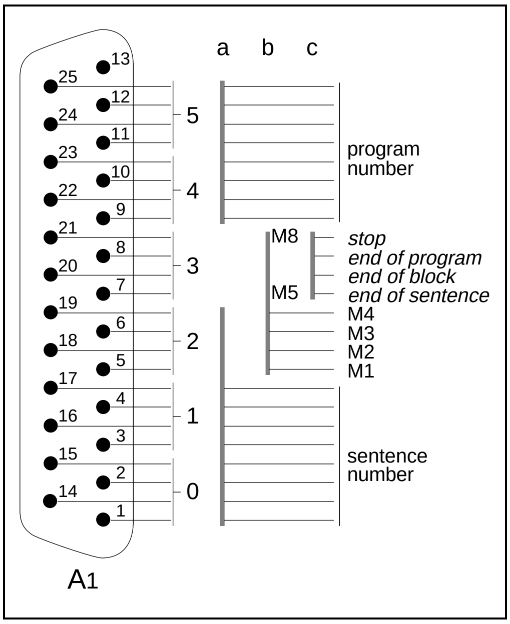
storage location	variant	meaning
2/**2**	2 »8 out1.2«	8 machine functions at decades 2 + 3
2/**8**	1 »output 1«	sentence and program number at decades 0...2, 4 + 5
2/**9**	3 »out 1.3«	program processing signals at decade 3

Output:

a) sentence and program number at decades 0 + 1 and 4 + 5; the 3rd digit of the sentence number at decade 2 must be zero

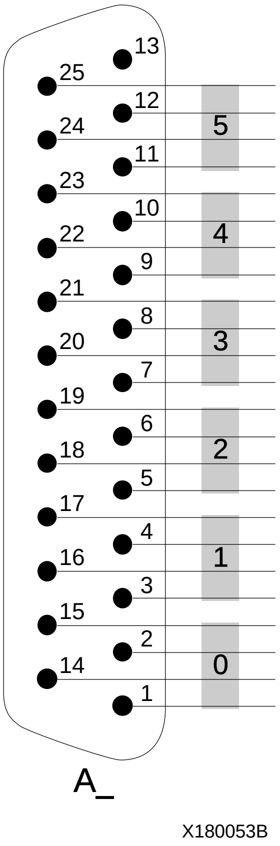
b) lower four machine functions M1...M4 at decade 2; the upper four machine functions M5...M8 at decade 2 must not be programmed (=0)

c) program processing signals at decade 3



E180053A

For your own entries:



4.11 Limit switches

4.11.1 Software limit switches and input monitoring

The positioning range can be limited by programming two special position values at the axis parameters:

- lower limit: 3/**71** (»Pos. min«)
- upper limit: 3/**72** (»Pos. max«)

The **»Pos. min« < »Pos. max«** condition must be observed. Otherwise, this causes a parameter error.

The programming of the limits effects the activation of the

– **input monitoring** of all absolute position values within the programming mode of nominal values and for the direct input modes,

– **software limit switch** function if storage location 3/**73** is programmed accordingly (refer to below).

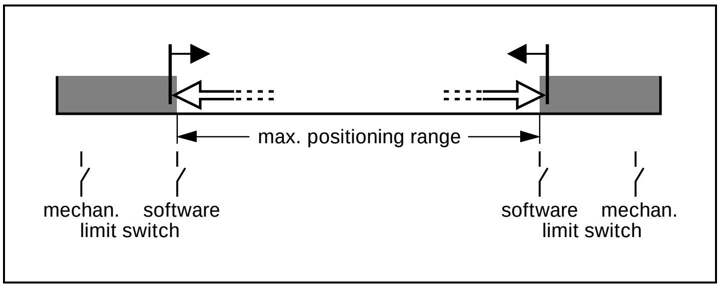
The actual position is controlled constantly during a positioning or parking process, a manual positioning, or a reference search routine if a 1 (»driving«) is programmed in storage location 3/**73**. Once the upper or lower limit is exceeded,

– a braking process is initiated,

– the */fault* signal is output at terminal $P_1$30 (level changes from High to Low),

– the separation points in display B are flashing (only visible if the corresponding axis is just displayed).

The drive can now be positioned only in the opposite direction.



E2800059

The same also applies if a 2 (»^ start«) is programmed at storage location 3/**73**. In addition, when specifying a *start* signal the Controller first tests if the nominal position of the new sentence would be beyond the limit values (this is principally only possible for processing nominal lengths or if the limits have been changed <u>after</u> programming of the nominal positions). In this case, the limit switch will trigger including the effects described above.

Independent of the programming of storage location 3/**73**, the limit switch function is always **deactivated**

– with 3/**71** (»Pos. min«) = 3/**72** (»Pos. max«) = 0,
– with rotary table positioning (refer to section 4.8),
– during the first calibration process provided that the variant 1 (»n.s.cal«) or 3 (»sec.cal.«) is programmed at storage location 1/**2** (power failure security).

4.11.2 Hardware limit switch (option)

The signals of mechanical limit switches can be applied to the Controller via one of the optional data input modules E1 ... E3 (GEL 8610: E4).

The following signal levels apply:

- **High** level ⇒ limit switch has not been activated (drive is within the admissible range)
- **Low** level ⇒ limit switch has been activated (drive is beyond the admissible range)

Starting from software version 12.00, the switching levels for the hardware limit switches are **inverted** (previously release at High level, now at Low level!).

If therefore, with an existing controller, the software version 11.xx is **replaced** by a newer version, any possibly built-in inverter circuits have to be removed or they now have to be inserted.

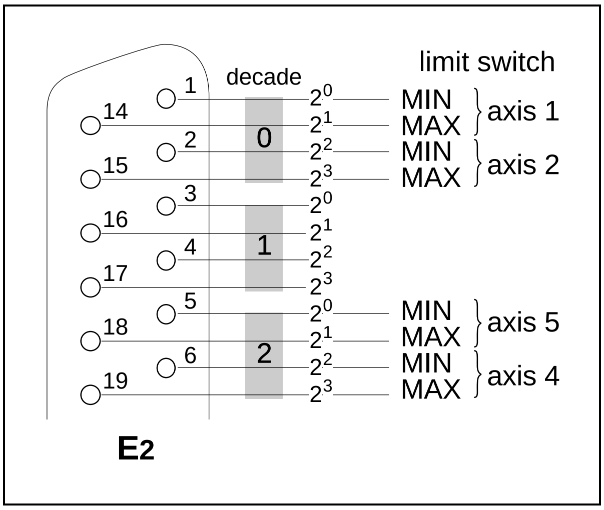
If a limit switch is triggered, the Controller reacts as if a software limit switch has been triggered (refer to the previous section). Furthermore, a corresponding error message is stored in the error memory (refer to section 6.2, error numbers 28 and 29).

The configuration of the signal inputs is determined via the axis parameter 3/**74** (refer to appendix A).

The two signals of any 2 axes can be grouped to one decade. The **numbers** of **both** axes must, however, **not be** even or odd! This means that the limit switch signals of the two axes 1 and 2 or 1 and 6 can be grouped together but not those of axes 1 and 3 (both uneven) or 2 and 4 (both even).

The axis with the **odd** number must always be connected to the **less significant** connections of the decade (2^0 and 2^1), with the signal for the lower counting limit (MIN) being connected to 2^0 or 2^2 and the one for the upper counting limit (MAX) to 2^1 or 2^3, respectively (refer to the following example).

Example: Limit switches of 1st and 2nd axes are connected to decade 0 of the E2 data input module and the ones of 4th and 5th axes to decade 2 of the same module

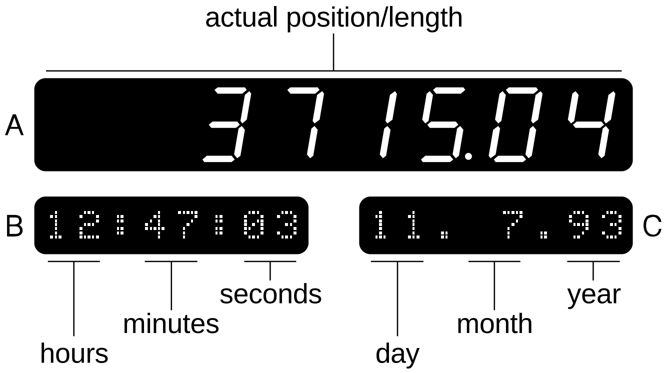


E180053C

4.12 Clock (option)

With the use of a serial interface (module V) with battery-buffered clock, it is possible to indicate the actual date and time as well as the time of the occurrence of a fault.

The actual date and time can be indicated in each operating state of the automatic mode (displays B/C) by pressing the [F1]+[⊙] key combination resp. the [⌚] key (GEL 8610):



E180077C

For the indication of **fault time values** refer to paragraph 6.2.

As long as the time is indicated, no other function can be activated at the Controller.

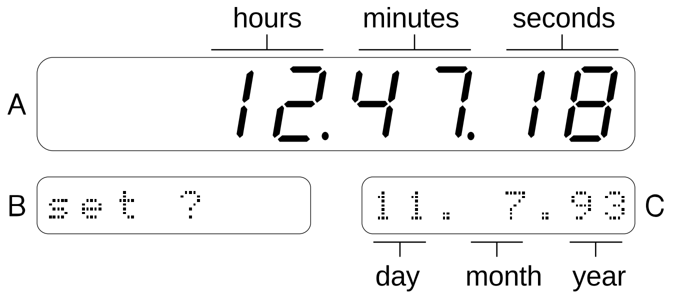
The clock function is terminated by means of the [ESC] key, then displays B and C again indicate the normal operating data.

Setting the clock

The clock is set within the programming mode of machine parameters, i.e. immediately after having activated the programming mode (after the password entry).

☛ Press the [F1]+[⊙] keys resp. [⌚]

Display A now indicates the time and display C the date:



E180077B

☛ Confirm the safety inquiry set ? in display B with [⇨]

The digits for the seconds are flashing now and the ? disappears

- Enter new second value and confirm it with or skip the input with . A wrong input can be cancelled by means of the key, the actual value will then be indicated again.

  The actual input position now shifts to minutes.

- Enter the values for minutes and hours as described above for the seconds

  After having entered the hours, the input position shifts to the calendar day.

- Enter calendar day, month and year, as described above

  After having entered the year, the function recommences.

- Effect further modifications as described above or quit the function with (the latter is possible at any time during the setting process)

4.13 Linear path control

The standard software of the Controllers GEL 8310 / 8610 contains a **linear path control** for all axes.

With the 'Circular interpolation' option being present (refer to appendix O), a **circular path** control can be activated additionally for the first two axes.

The linear path control can be activated for the following units/axes:

- unit **1**: axes 1 to 3 (GEL 8610: 6)
- unit **2**: axes 2 and 3

For this, the variant 1 (»linear«) must be selected for storage location 2/**5** of the corresponding unit.

The **rotary table positioning must not** be activated when using the path control (storage location 3/**8** = 0 for all axes). Otherwise, a `Param. error` is caused.

There are two methods of presetting the **speed rate** for the combined axes:

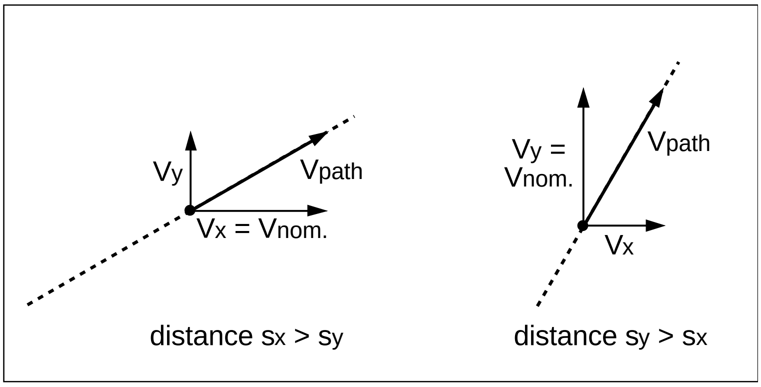
- separately for each sentence

  In this case, the 'speed rate' nominal value type must be part of the sentence, determined at storage location 2/**3**.

  The input format or the measuring unit for the speed rate is fixed by the **first** axis of the unit (storage locations 3/**45** and 3/**46**; refer to section 4.2).
- identically for all sentences

  In this case, the 'speed rate' nominal value type is not part of the sentence. The value is preset via the axis parameter 3/**34** of that axis that has to travel the longer distance.

The **path speed** consists of the speed components of the combined axes. The axis that has to travel the longer distance is always responsible for presetting the speed value. The speed rate(s) of the other axis (axes) is (are) adapted accordingly, depending on the path angle. The following diagram shows an example for two axes (left: axis 1 = X travels the longer distance s_x; right: axis 2 = Y travels the longer distance s_y):



E180093A

The parameters of the axis with the longer distance are also used for controlling the other axes being part of the linear path control. Therefore, the parameters

- maximum speed (v_{max}, storage location 3/**32**),
- working speed (v, storage location 3/**34**) as well as
- accelerating/braking (t_{accel}, t_{brake}, storage locations 3/**35**...**38**)

should be programmed equally for all axes with the values of that axis which possesses the most unfavourable parameters.

The sentences may be **continuously** processed (positioning without stop), refer to section 4.3 and the following section.

Spline

A spline function can be used to avoid jerky speed rate changes (infinite acceleration) with path controlled axes in the case of angular paths and continuous sentence processing. Herewith, the maximum appearing acceleration for one (slave) drive is limited to the sum of the accelerations of two drives (master and slave; here, master is always the axis that has to travel the longer distance).

Activation:

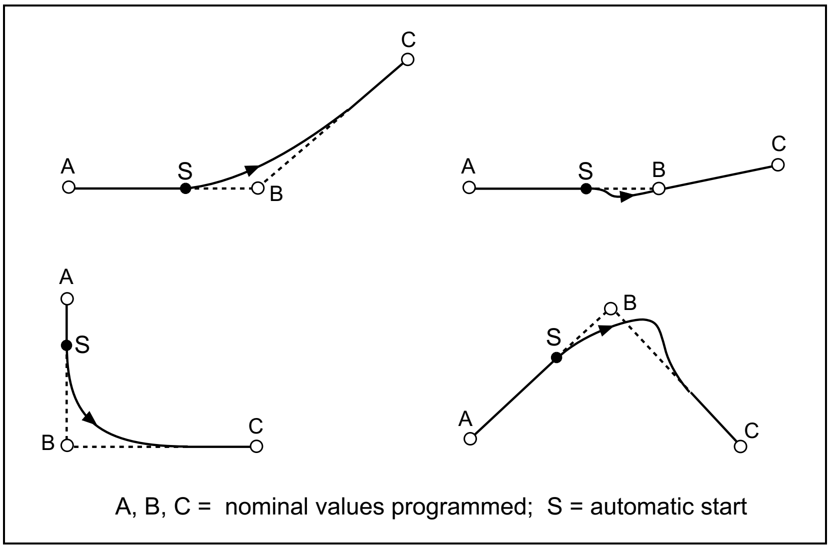
- storage location 2/**3** = 3 (»yes (➡3)«)

  precondition: storage location 2/**5** = 1 (»linear«). Otherwise, a `Param. error` is caused.

  The internal start is generated like for mode 1 of the continuous sentence processing (refer to section 4.3).

The spline function causes the (theoretical) path to be exited in the area of the knee-points more or less far, depending on the angle of the path change and on the processing speed as well as the programmed maximum values for acceleration and speed rate of the involved axes. The restriction described in section 4.3 is not valid for this function.

The following figure shows some typical path courses:



E180093G

4.14 Program flow instructions

Within the programming mode of nominal values, the following additional control functions are available:

instruction	key combination	meaning
CALL Pr.	F3 + 1	process another program
JUMP Pr.	F3 + 2	resume to work with another program
JMP sent	F3 + 3	continue processing at another sentence
IF I/O	F3 + 4	resume program processing depending on the state at a certain signal input or output

General features:

- The instructions can only be activated at the **beginning of a sentence** and occupy an entire sentence (first and only nominal value, the same as the end of program). The error message `Only for s.begin` is output if an instruction is programmed at a sentence position other than the first one.
- The instructions can not be **activated/deactivated** by the repeated actuation of the corresponding key combination as this is the case at the change-over between position and length. A change-over between the different instructions is, however, possible.
- If an instruction is to be **cancelled** or to be **replaced** by a ‘normal’ sentence, the corresponding instruction sentence is first to be deleted (F3 + [delete key] resp. [delete key]). When inserting a normal sentence (using F3 + [insert key] resp. [insert key]) then the sentence currently being active must also be a normal one, i.e., no ‘instruction’ sentence. Alternatively, a ‘normal’ sentence can be substituted or inserted at the desired position by copying ((F3 + 8 resp. [Copy], see section 3.2.1).
- Branchings or jumps are only possible **inside the unit**.
- Each program control instruction causes the generation of an **internal start signal** regardless of whether or not auto start (storage location 2/**10**) has been activated. This means that a program will automatically continue until either a positioning instruction (nominal position/length) or the end of the program is reached.
- A **timer function** for the delayed processing can be simulated by programming an auto start time for each sentence (refer to storage location 2/**1**).

- With a **serial data transmission** by means of the LB2 protocol, dummy values must be transferred after the code for the respective instruction (see there) for all nominal value types that are contained in a normal sentence (e.g. piece number, determined by the programmed sentence structure of the unit) even if they have no meaning in this case (they are undefined).

4.14.1 Call subroutine (CALL Pr.)

If the Controller encounters the CALL Pr. instruction while processing a program it will temporarily branch to the program with the indicated number (beginning with sentence no. 1). This program is then executed as often as specified by its cycles value. After the last program execution, processing of the original program will be continued starting at the sentence following the CALL instruction.

Up to **20** CALL instructions can be **nested.** If the number is exceeded, the program is interrupted at the corresponding position and the error # 32 (Too many CALLs) is stored in the fault memory (can be retrieved using [F1]+[0]).

If another program is called from a subroutine (CALL) with JUMP then after processing the other program a jump is made back to the program that originally contained the CALL instruction (processing continues with the sentence following the CALL instruction). A subroutine is consequently terminated by a program jump (JUMP Pr.) included in the subroutine.

Input:

- ☛ [F3]+[1]
- ☛ *program number* (1 ... 99)
- ☛ [⇨]

Upon the input, the existence of the specified program is not verified. If the program does not exist the current program is aborted when the CALL instruction is executed (controller is switched over to the reset state) and the error # 20 (Invalid program) is stored in the fault memory.

Coding for serial transmission (length = 4 bytes):

byte:	4 (MSB)	3	2	1 (LSB)
	10h	00h	00h	**prg. no.**

4.14.2 Jump instructions (JUMP Pr., JMP sent)

If the Controller encounters the JUMP Pr. instruction while processing a program it will directly branch to the program with the indicated number (processing starts with sentence no. 1). With JMP sent processing will resume at the indicated sentence, thus skipping certain sentences.

In order to avoid endless loops, the number of **succeeding** JUMP instructions is limited to 5. If this number is exceeded, the program is interrupted at the corresponding position and the error # 31 (More as 5 JUMPs) is stored in the fault memory (can be retrieved with (F1)+(0)). CALL instructions do not influence the counter of JUMP instructions which means that they do not interrupt a JUMP chain. This counter is, however, reset to zero by a positioning process or an IF instruction.

If another program is called from a subroutine (CALL) with JUMP Pr. then after processing the other program a jump is made back to the program that originally contains the CALL instruction (processing continues with the sentence following the CALL instruction).

Input:

JUMP Pr.	JMP sent
☛ (F3)+(2)	☛ (F3)+(3)
☛ *program no.* (1 ... 99)	☛ *sentence no.* (1 ... 999)
☛ (⇨)	☛ (⇨)

Upon the entry, the existence of the specified program or sentence is not verified. If the program or sentence does not exist the current program is aborted when the JUMP instruction is executed (controller is switched over to the reset state) and the error # 20 (Invalid program) and/or error # 30 (Invalid sentence) is stored in the fault memory.

Coding for serial transmission (length = 4 bytes):

byte:	4 (MSB)	3	2	1 (LSB)
JUMP Pr.	**20**h	00h	00h	**prg. no.**
JMP sent	**40**h	00h	**sentence no.**	

4.14.3 Signal-dependent branching (IF I/O)

The program can be conditionally executed depending on the signal state at certain inputs and outputs (I/O) of the Controller.

If the condition is true, i.e. the signal state at the called input/output is **logically 1** (High level), the program is continued with the **sentence immediately** following the IF instruction. Otherwise (logically 0), the program is continued with the next sentence plus one, i.e. the sentence following the IF instruction is skipped.

Output signals are internally scanned. This means that all kinds of output signals may be used without the pertaining data output having to be physically existing. Consequently, it is possible, e.g., to assign program execution signals or machine functions to a fictitious data output A3 and subsequently to scan it. When calling **input signals,** the corresponding data input connector must of course be available. However, an assignment of a function is not required.

When starting from the reset state, it is not possible first to process an IF instruction (directly in the first sentence of the started program or via a CALL/JUMP instruction from another program just started) so as to avoid non-defined initial positions. In this case, the program is immediately aborted (in a way, the Controller remains in the reset state). Error # 33 (`IF from reset`) is stored in the fault memory (can be retrieved with [F1]+[0]). Consequently, after a start at least one positioning instruction must precede an IF instruction.

By chaining several IF conditions, logical ANDs or ORs can be implemented (refer to the example in the following paragraph).

Input:

- ☛ [F3]+[4]
- ☛ *I/O number* (s. b.)
- ☛ [⇨]

The **I/O numbers** are coded as follows:

1xx ... **4**xx	→	data input **E1** ... **E4**
5xx ... **8**xx	→	data output **A1** ... **A4**
xx = 00 ... 23	→	bit number 0 (2^0) ... 23 (2^{23}) *
90x	→	***actual = nominal*** signal
100x	→	***reference reached*** signal
x = 1 ... 6	→	axis 1 ... 6

* Normally the data inputs/outputs are grouped into decades (counting from bit 2^0 to 2^3 each), refer to appendix B. Concerning the IF instruction, a binary devision is performed, i.e. counting is continued with decade 1, with bit 2^4 instead of 2^0, up to 2^{23} (= decade 5, bit 2^3).

The entry is monitored concerning valid value range. In case of an error, the `Invalid entry` message is output (confirm with any key).

Coding for serial transmission (length = 4 bytes):

byte:	4 (MSB)	3	2	1 (LSB)
	30h	00h	**I/O no.**	

4.14.4 Examples

1. Logical OR

In case of High level at position 2^0 **or** 2^1 of the 2nd data input, program 2 is to be processed, otherwise program 3.

Program 1 (control program):

sentence 1:	`Pos. A 1`	0	initial position
sentence 2:	`IF I/O`	200	bit 2^0 = 1?
sentence 3:	`JMP sent`	8	yes (⇒ process program 2)
sentence 4:	`IF I/O`	201	no; bit 2^1 = 1?
sentence 5:	`JMP sent`	8	yes (⇒ process program 2)
sentence 6:	`CALL Pr.`	3	no; process program 3
sentence 7:	`JMP sent`	2	ready; back to the first scan
sentence 8:	`CALL Pr.`	2	process program 2
sentence 9:	`JMP sent`	2	ready; back to the first scan
sentence 10:	`Cycles`	0	end of program

2. Logical AND

Only if High level is present at position 2° **and** 2^1 of the 2nd data input, program 2 is to be processed, otherwise program 3.

Program 1 (control program):

sentence 1:	Pos. A 1	0	initial position
sentence 2:	IF I/O	200	bit $2^0 = 1$?
sentence 3:	JMP sent	5	yes (⇒ next scan)
sentence 4:	JMP sent	7	no (⇒ process program 3)
sentence 5:	IF I/O	201	bit $2^1 = 1$?
sentence 6:	JMP sent	9	yes (⇒ process program 2)
sentence 7:	CALL Pr.	3	no; process program 3
sentence 8:	JMP sent	2	ready; back to the first scan
sentence 9:	CALL Pr.	2	process program 2
sentence 10:	JMP sent	2	ready; back to the first scan
sentence 11:	Cycles	0	end of program

3. Simple positioning sequence (principle)

Material of differing length is to be pushed forward in a slide, providing the material with borings at a distance of optionally 1000 or 2000, depending on a signal at the 2nd data input, position 2^1 (1 = length 2000). Using another signal (2nd data input, position 2^2) the end of the material is determined and the slide is move to position 0. The sentences include a machine function (output with *actual=nominal).* They are automatically started (e.g. every 5 sec). The process enable is scanned in a waiting loop (signal at position 2^0 of the 2nd data input, 1 = start enabled).

Program 1:

sentence 1:	Pos. A 1 M. funct	0 0	beginning of material do not drill
sentence 2:	IF I/O	200	start permitted?
sentence 3:	JMP sent	5	yes (⇒ next scan)
sentence 4:	JMP sent	2	no; repeated scan (waiting loop)
sentence 5:	IF I/O	202	end of material?
sentence 6:	JMP sent	1	yes (⇒ initial position)
sentence 7:	IF I/O	201	no; distance 2000?
sentence 8:	JMP sent	11	yes (⇒ distance = 2000)
sentence 9:	Leng.A 1 M. funct	1000 1	no; distance = 1000 drill
sentence 10:	JMP sent	2	ready; back to first scan
sentence 11:	Leng.A 1 M. funct	2000 1	distance = 2000 drill
sentence 12:	JMP sent	2	ready; back to first scan
sentence 13:	Cycles	0	end of program

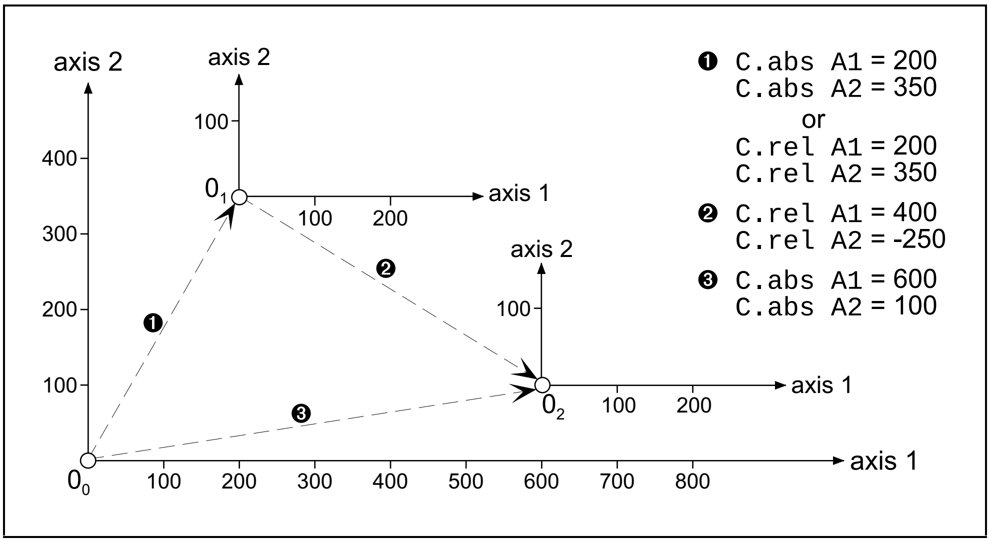
4.15 Coordinates offset

With units containing only ‘pure’ positioning axes, the origin of coordinates can be offset for each sentence, namely either

– **absolutely** with reference to the machine zero point (angle encoders zero or reference point with incremental systems) or

– **relatively** to the just valid origin.

The following diagram demonstrates these possibilities.



E180039B

If a sentence is started with a coordinates offset, only the **actual value counter** of the concerned axes is **changed**: **No** positioning process is involved. Then, the next sentence is started with an internally generated signal. If it contains the nominal **length 0** for all axes then here also no positioning is performed.

Requirements for all involved axes:

- system of absolute dimensions (storage location 3/**44** = 0 [»reference«])
- no rotary table function (storage location 3/**8** = 0)

Activation:

☛ press at the beginning of a sentence (instead of the position for the first axis in the unit) the key combination

– (F1)+(1) for an **absolute** offset of the origin
(display C: `C.abs Ax`, *x* = no. of the axis) or

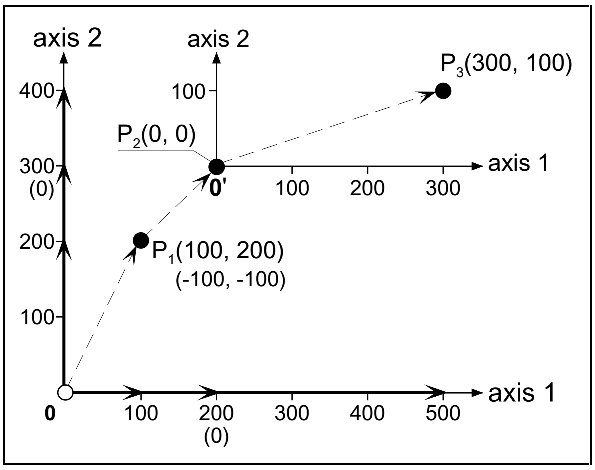
– (F1)+(2) for a **relative** offset of the origin
(display C: `C.rel Ax`, *x* = no. of the axis)

☛ Enter the desired absolute or relative zero coordinate for the *x* axis (display A).

The input features for the coordinates offset correspond to the program flow instructions (refer to section 4.14, General features).

<u>Positioning example</u> for an unit consisting of two axes:

sentence 1:	Pos. A 1 Pos. A 2	100 200	coordinates = **machine** coordinates
sentence 2:	C.abs A1 C.abs A2	200 300	coordinates offset absolute to the machine zero point
sentence 3:	Pos. A 1 Pos. A 2	0 0	coordinates = **work piece** coordinates
sentence 4:	Pos. A 1 Pos. A 2	300 100	
sentence 5:	...	...	



E180039C

- With the first starting signal position P_1 is moved to.
- With the second starting signal, the actual values of axis 1 and axis 2 are set to -100 (coordinates offset) and then with an internal start, the position P_2 is moved to (with reference to the machine zero point the coordinates would be (200, 300)).
- Position P_3 is moved to with the third starting signal.

The machine zero point is reactivated with the next reset signal, i.e. the displayed actual values are related to this. (If, in the example above, the Controller is reset in P_3 then the actual value of axis 1 is set to 500 and the one of axis 2 to 400.)

1. Programmed positions of **software limit switches** (storage locations 3/**71**…**73**) are related to the actual positions in the **current coordinate system**. In an unfavourable case, with coordinates offset **monitoring** can thus become **ineffective**!
2. When using **absolute encoders**, the end of the counting range can be reached by one or several unfavourable coordinates offsets. Thus, a **prohibited actual value jump** can appear during positioning.

Tip

To cut, e.g., individual work pieces from material, the data being valid for all work pieces such as (relative) positions or lengths and other sentence elements (e.g. speed rate with continuous sentence processing) could be stored in a program no. 2 while the program no. 1 to be started contains successively the coordinates offsets for each work piece separated by the 'Call Pr. = 2' program flow instruction (= single processing of the work piece program no. 2).

Coding for serial transmission (length = 4 bytes):

byte:	4 (MSB)	3	2	1 (LSB)
absolute	**E**xh	xxh	xxh	xxh
relative	**D**xh	xxh	xxh	xxh

value of the offset (negative: two's complement)

If a 'normal' sentence contains further elements (such as e.g. the number of pieces), then the values for these will (must) be transferred following the coordinates offset, even if they have no meaning in this case (they are undefined). There are always so many nominal values per sentence to be transferred as they are determined by the programmed sentence structure of the unit (exception: last sentence which contains the program end).

5 Commissioning

The procedure specified in the following for the commissioning of the Controllers GEL 8310 / 8610 can only be a proposal and is just one of various possibilities. Use your experience to optimize this procedure for your special needs and applications.

Commissioning basically includes the following steps:

❏ Standard settings (section 5.1)

❏ Actual value adjustment (section 5.2)

❏ Adaptation of the analogue output (sections 5.3 to 5.5)

❏ Optimization of control parameters (section 5.6)

❏ Controller configuration (section 5.7)

The last four items must be carried out for each connected axis.

The table of contents on the previous page provides a **survey** on the commissioning **procedure**. If possible, do observe the order. Paragraphs describing a programming procedure or setting are marked with »✔«.

Prerequisites, initial state

❍ You have already made yourself familiar with the operating and the functions of the Controller by reading this manual.

❍ The electrical connections to the peripheral equipment (encoders, drive amplifiers, initiators etc.) have already been established according to the connection diagrams in appendix B. Motor, amplifier and tachometer have already been matched with each other.

❍ Supply voltages are available.

❍ The machine part to be positioned should be approx. in the middle of the positioning area.

❍ An **EMERGENCY STOP** (for the drive) must be freely and quickly accessible (refer to section 4.1.3).

❍ All storage locations of the Controller are reset to 0 (default), except the system parameter 1/**3** (initial state as set at the factory); if this is not the case clear the storages (see sections 3.2.1 and 3.3.1, keys F3+7).

❍ All modules are configured properly, i.e., jumpers are set according to the requirements (see appendix B).

Does your Controller get its control signals from a PLC or similar automation devices?

If so, please check already during trial service whether there are always the same errors being stored in the fault memory of the Controller (cf. section 6.2). Responsible for that could be the control device generating several *start* signals per second which do not cause a start because of the *reset* signal still being active or the */stop* input being at Low level. Then each time the error # 14 or 13 is written to the EEPROM. This extended account of write cycles will shorten the service life of the EEPROM (about 10,000 write cycles at maximum)!

For this reason, please provide validity checks of the corresponding signal states in your control device program. Otherwise, the EEPROM – and, consequently, the Controller – will break down after a relatively short operational time with the display showing `EEPROM error`.

5.1 Standard settings

✔ Activate the programming mode for machine parameters: [F2]+[1], [9][2][2][8], [⇨].

System parameters

✔ **Storage** locations 1/**3** to 1/**5** (GEL 8610: 1/**8**): units

Combine the axes available into units according to the installation requirements. Confirm the security inquiry `Delete unit?` issued upon reprogramming by pressing the [⇨] key.

When commissioning the individual axes in the Automatic mode, these must be shown in the display individually:

[F3]+[0] [Unit]	to select the **unit**	[F3]+[⇧]/[⇩] [Axis+1]/[Axis−1]	to select the **axis** within the unit

Furthermore, for the axes to be commissioned the pertaining unit control inputs */stop* (High level) and *reset* must be connected (see appendix B).

Unit parameters

For the moment the unit parameters are not yet to be programmed.

Axis parameters

The axis parameters are dealt with in the following sections.

5.2 Actual value adjustment

5.2.1 Incremental encoders

✔ Storage location 3/**1**: edge evaluation

The type of edge evaluation for the encoder pulses depends on the actual measuring unit used in the installation (refer to section 4.1.1).

Example: existing encoder with 250 pulses/revolution, desired are 1000 increments/revolution ⇒ 4fold edge evaluation: »Incr. x4«

✔ Storage location 3/**2**: count direction

The correct count direction of the encoder will be set later on, as soon as the machine can be displaced (section 5.4).

✔ Storage location 3/**3**: multiplier

The edge-evaluated encoder pulses are multiplied by this value, before they are shown as count pulses in the actual value display.

Example: actual = 1000 increments/revolution, desired = 820 increments/revolution ⇒ multiplier = 0.82

✔ Storage location 3/**5**: decimal point

The number of decimals depends on the requested resolution.

Example: actual = 820 DispU, required = 82.0 actual measuring units (mm) with a resolution of 1/10 mm ⇒ decimals = 1 ≙ »X.X«

Now the adjustment of the actual value for the incremental encoder of the treated axis is completed for the time being.

5.2.2 Absolute encoders

✔ Storage location 3/**1**: code/ type of logic

If you do not know the encoder's **logic** type (with Gray or BCD code), you may find it out later on (see further below).

✔ Storage location 3/**2**: count direction

The encoder's correct count direction will be set later on, i.e., as soon as the machine can be displaced (section 5.4).

✔ Storage location 3/**3**: multiplier

The position transmitted by the encoder is multiplied by this value before it is shown in the actual value display.

Example: At one revolution the position value changes by 1024, however, a value of 1000 should be the result ⇒ multiplier = 0.9766, rounded up; the rounding error results in a respective positioning inaccuracy after several revolutions!

✔ Storage location 3/**5**: decimal point

The number of decimals depends on the requested resolution.

Example: actual = 820 DispU, required = 82.0 actual measuring units (mm) with a resolution of 1/10 mm ⇒ decimals = 1 ≙ »X.X«

✔ Storage location 3/**54**: resolution

In this case resolution means the bit range of the connected encoder (singleturn part + multiturn part = used parallel inputs of terminal strip W or Y; singleturn part for SSI encoders, connector S).

✔ Storage location 3/**55**: *enable* signal (terminal strip W/Y)

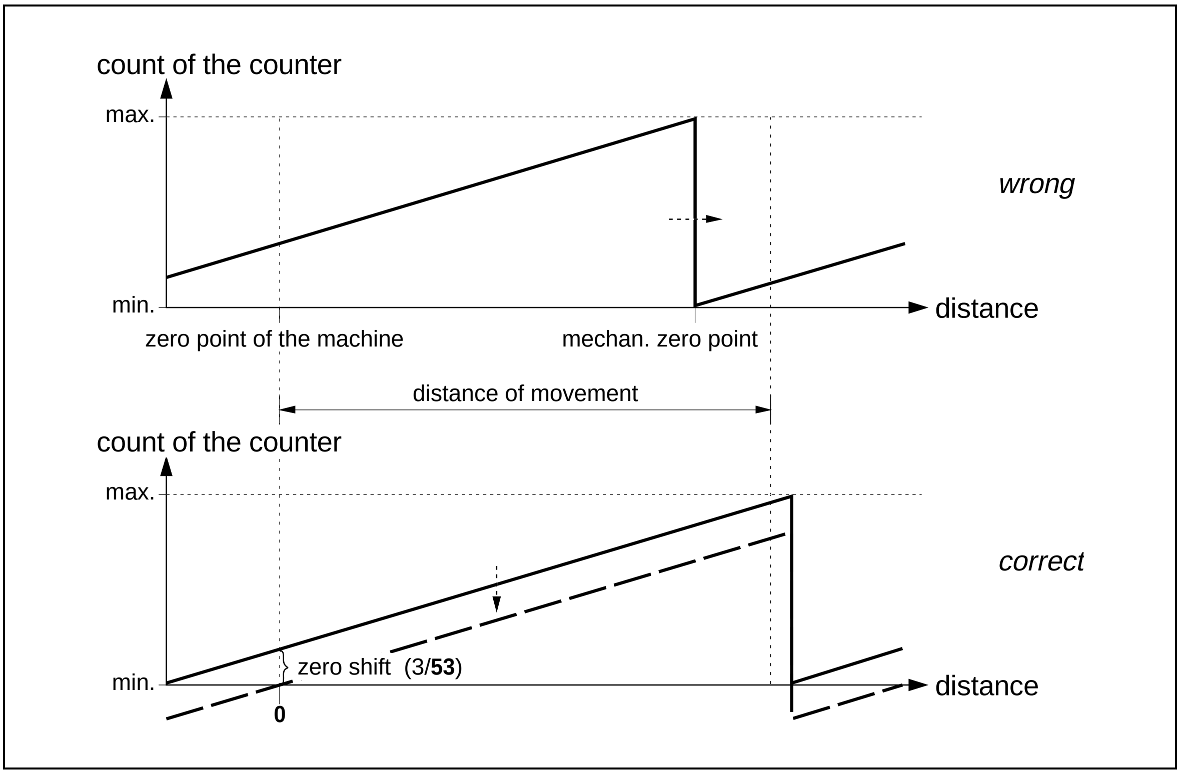
Set the logic level of the signal.

✔ Determination of the encoder's type of **logic** (for Gray or BCD code)

If the encoder has not yet been attached to the drive or may easily be dismounted, turn the encoder axis and observe the actual value display: does the actual value change **erratically**, the **logic reverse** must be programmed. If this possibility of testing is not given, checking must be performed as soon as the machine can be displaced (see section 5.4). This requires, however, that the mechanical zero point of the encoder is outside the displacement area (see next item).

✔ Mechanical zero point

The mechanical zero point of an angle encoder should **not be inside** the area of displacement. Should this be the case after all, the actual value counter would jump from the maximum to the minimum value during the positioning process or vice versa, thus confusing the control completely. Therefore, the axis of the uncoupled angle encoder must be turned accordingly (see following figure and example).



E180039A

Example: angle encoder with a resolution of 1024×512 (19 bits), counting range: 524,288 DispU (multiplier = 0, no decimal point)

mechanical displacement: 307,200 DispU ≙ 300 revolutions (1,024 DispU ∗ 300)

Count of the counter in the machine's zero point: 250,238 DispU

Count of the counter at the end of the displacement area:
250,238 + 307,200 DispU = 557,438 DispU
This value, however, is above the counting range and that means: the mechanical zero point of the angle encoder is located inside the displacement area!

Measure: the axis of the angle encoder must be turned back by **at least** 33,150 DispU; then the count display would read 217,088 DispU in the machine's zero point and the mechanical zero point of the angle encoder is right at the end of the displacement area (217,088 + 307,200 DispU = 524,288 DispU).

✔ Storage location 3/**53**: zero shift

Starting from the above example, here the value –217,088 should be entered to ensure that in the machine's zero point the actual value counter indicates 0.

Now the actual value adjustment for the absolute encoder of the treated axis is competed.

5.3 Preparations for displacing the drive

✔ Storage location 3/**19**: manual drive control

Fix here whether the drive should be controlled via the keyboard or an external data input. In the second case, the module and the position where the control signals should be read must be determined.

✔ Storage locations 3/**21** and 3/**23**: slow speed forward and reverse

To ensure a safe operation these values are first limited to 1 DispU/sec (1 or 0.1 or 0.01, depending on the set resolution).

✔ Storage location 3/**26**: voltage range of the analogue output

Fix here whether the displacement voltage should be bipolar or unipolar.

✔ Storage location 3/**31**: maximum voltage value U_{max}

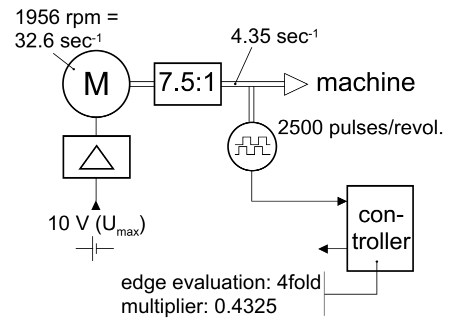
This value is determined by the maximum speed of the drive v_{max} (system-dependent, e.g. 8.50 V for v_{max} = 3000 rpm).

✔ Storage location 3/**32**: maximum speed rate v_{max}

This is a installation-specific value and indicates the speed in actual measuring units per second at output voltage U_{max}.

For the moment, it is sufficient to indicate an estimated value or a value calculated from the installation data. Later on the value will be optimized (refer to section 5.6.1).

Example:



$v_{max} = 4.35\ sec^{-1} \quad 2500\ pulses \quad 4 \quad 0.4325 =$
18814 pulses/sec

E010H

✔ Storage location 3/**33**: control factor K_{sp}

By programming a value for the control factor (which is very small for the beginning) the closed-loop control is activated: K_{sp} = **1.0**.

✔ Storage location 3/**34**: working speed rate

For the moment, program a value which is near to v_{max} (approx. 95%).

✔ Storage locations 3/**35** to 3/**38**: values for accelerating and braking

These data should already be available from adapting the drive to the machine. Should this not be the case, values must first be estimated, e.g.: how much time takes the drive for accelerating from standstill up to the maximum speed ($t_{accel+/-}$) and for braking from the maximum speed down to standstill ($t_{brake+/-}$)? These values should rather be too large than too small. The mathematical determination of the parameters from the drive data will not be described at this place. The assessed values will still be optimized at a later time (by means of a current monitor at the drive amplifier or an oscilloscope, refer also to section 5.6.3).

Note: Values must not be programmed for all four times (there are even some options not allowing this, see corresponding description in appendix O). When setting to 0 the value of another time parameter is used:

$t_{brake-} = t_{accel-}$ for 3/**38** = 0 $t_{brake+} = t_{accel+}$ for 3/**37** = 0
$t_{accel-} = t_{accel+}$ for 3/**36** = 0

✔ Storage location 3/**42** and 3/**43**: contouring error $S_{max+/-}$

A maximum contouring error is to be determined for safety reasons so that the drive can be stopped if a wrong travel direction has been set for the control (refer to next section). The values should be about 10% of the entire displacement area. They will be reduced later according to the prevailing operating conditions.

✔ Storage location 3/**50**: control for manual displacement

For the following settings the closed-loop control needs to be activated: 3/**50** = 1 (»active«).

WARNING

The **position loop control** in the interrupted or reset condition must **not** yet been activated at that moment (storage location 3/**47** = 0) in order to prevent racing of the drive in case of a wrong travel direction assignment during this mode of operation (also refer to section 4.1.3).

✔ Exit the programming mode for machine parameters ([F2]+[0] resp. [Auto]).

Next the count direction of the encoder and the polarity of the voltage output must be checked.

5.4 Count direction and voltage polarity

✔ Ensure that the axis to be adjusted is displayed (see section 5.1).

✔ Displace the drive manually: slow speed **forward** ("**>**"). The actual value (display A) must **count up**. Also check in which direction the machine moves.

WARNING

If the actual value **counts down**, **immediately abort** the manual displacement.

Because of the programmed small value for slow speed the drive will not start moving immediately, but only after the slowly increasing voltage having reached a certain value at the analogue output (U_{min}). Then the drive runs at a speed of 1 DispU/sec provided that the count direction is correct. Otherwise, **the speed increases continuously**.

✔ Proceed as follows:

➨ If the actual value counts **up** and the machine, however, does **not** move **forward** as requested,

– exchange the count connections 0°/90° of an incremental encoder at the Controller <u>or</u>

– reverse the count direction (storage location 3/**2**)

and

– exchange the connections of the amplifier input or analogue output (only feasible, if the amplifier inputs of the several axes are **not** inter-connected through a joint earth line, see terminal strip D in appendix B) <u>or</u>

– reverse the polarity of the analogue output (storage location 3/**25**).

➨ If the actual value counts **down** and the machine, however, moves **forward** as requested,

– exchange the count connections 0°/90° of an incremental encoder at the Controller <u>or</u>

– reverse the count direction (storage location 3/**2**).

➨ If the actual value counts **down** and the machine does **not** move **forward** as requested,

– exchange the connections of the amplifier input or analogue output (only feasible, if the amplifier inputs of the axes are not intercon-nected through a joint earth line, see terminal strip D in appendix B) <u>or</u>

– reverse the polarity of the analogue output (storage location 3/**25**).

➨ If the actual value changes **erratically** (only possible for absolute encoders with BCD or Gray code), select the appropriate code type on storage location 3/**1** with **logic reverse**.

The next step is to determine the minimum voltage values for the drive amplifier which is necessary to start the drive in the forward and reverse direction (if not yet known).

5.5 Minimum voltages

The following experimental determination assumes, that there are threshold voltages U_{min} > 1 mV for the amplifier used whose exact values are not known. The Controller-specific forward direction ("**>**") is here assigned a positive voltage. This assignment may have been modified according to the description in the previous section. In this case, replace the word ‘forward’ by ‘reverse’ and vice versa in the following description.

U_{min+}

✔ Ensure that the axis to be adjusted is displayed.

✔ The output voltage must be shown in the display (select with [⬇]/[⬆]).

✔ Displace the drive manually: slow speed **forward** ("**>**", [F1]+[⬇])

Now you will observe a slow increase of the (positive) voltage, the actual value in the display staying constant. At a certain voltage value the drive will start moving forward, which you will recognize by the fact that the actual value display changes by 1 increment/sec (= 1 DispU/sec). Remember the voltage value and repeat this procedure once or twice.

✔ Return to the programming mode for machine parameters.

✔ Storage location 3/**29**: minimum positive voltage value U_{min+}

Enter the largest of the remembered voltage values <u>without</u> operational sign.

U_{min-}

✔ Repeat the procedure described above, however, with slow speed in **reverse** direction ("**<**", [F1]+[⬆])

Tip: If there is an enormous difference between the two values U_{min+} and U_{min-}, it might be advantageous to readjust the offset at the amplifier and to determine the voltage values anew.

✔ Storage location 3/**30**: minimum negative voltage value U_{min-}

Enter the largest of the remembered voltage values <u>without</u> operational sign.

The next step is the optimization of the coarsely adjusted control parameters.

5.6 Optimization of control parameters

5.6.1 Maximum speed

To check or optimize v_{max} the drive must be displaceable for a sufficient period of time (e.g. 5 seconds), so that the output voltage for the set (working) speed rate and also the contouring distance (»Delta_s«) can be read in the display.

✔ Storage locations 3/**21** and 3/**23**: slow speed forward and reverse

Now values are entered for the slow speed rate which are in the order of the later **positioning** speed rate (in actual measuring units per second); the definite values for the **manual** displacement control will be determined later (section 5.7).

If this speed cannot be maintained long enough (because the displacement area is too small) smaller values must be programmed, accordingly. It must, however, be considered in this connection that potential nonlinearities in the drive system can lead to major control deviations when positioning later on.

✔ Exit the programming mode for machine parameters.

✔ Ensure that the axis to be adjusted is displayed.

✔ The display must show the output voltage (select with ⬇/⬆).

✔ Move the drive – at slow speed forward or reverse – until the voltage value (U) belonging to the (working) speed (v) is constant. Remember this value.

Calculation of v_{max} as per following formula:

$$v_{max} = v * \frac{U_{max} - U_{min+}}{U - U_{min+}}$$

v_{max}	→	storage location	3/**32**
v	:	"	3/**21**, 3/**23**
U_{max}	:	"	3/**31**
U_{min+}	:	"	3/**29**
U	:	display C	

✔ Storage location 3/**32**: maximum voltage v_{max}

If the new calculated value differs strongly from the one programmed previously, determine also the ratio of both values. The value programmed for the working speed (storage location 3/**34**) must be adjusted in the same ratio.

✔ Perform check runs: the contouring distance (Delta_s, select for display) is now clearly lower in the range of constant speed. (An additional reduction may be obtained via the control factor K_{sp}, see next section.)

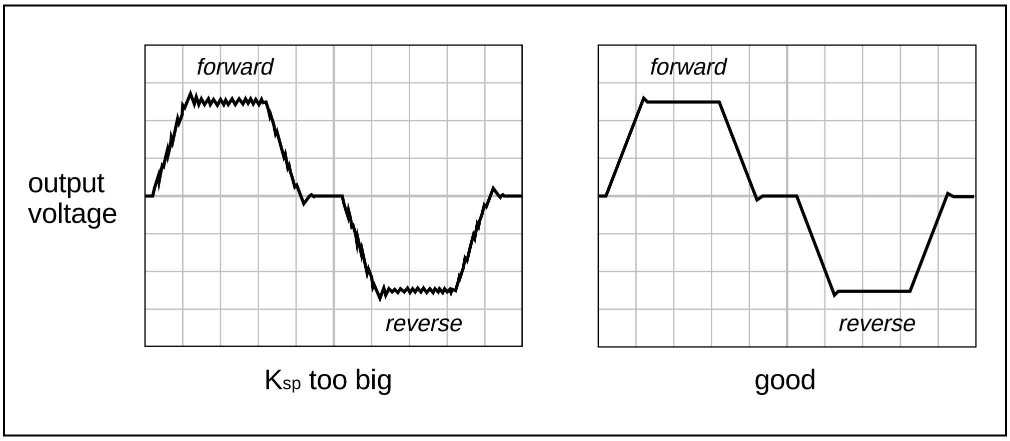
5.6.2 Control factor

The purpose of this setting is to make the control dynamics as big as possible via the factor K_{sp}, but not to make it too sensitive. It is advisable to use an oscilloscope for the process of optimization to examine the voltage at the analogue output (terminal strip D).

✔ Storage location 3/**33**: control factor K_{sp}

Increase the value step by step, i.e., depending on the dynamics and load of the drive by 0.1% to 0.5% of the used maximum speed rate (e.g. for v_{max} = 5000 actual measuring units/sec $\rightarrow$ ΔK_{sp} = 1.0 ... 5.0).

Check by means of the oscillogram, whether huntings occur during the entire displacement process. In the affirmative, reduce the control factor accordingly:

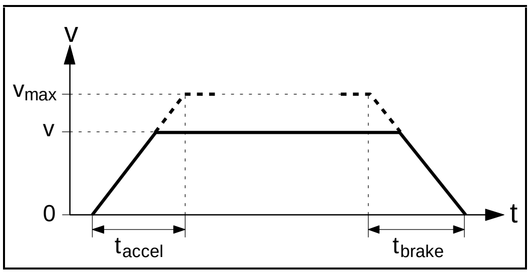


E180016C

Tip: An unstable control behaviour may also be recognized by the displayed value for the output voltage oscillating more intensively when travelling at constant speed.

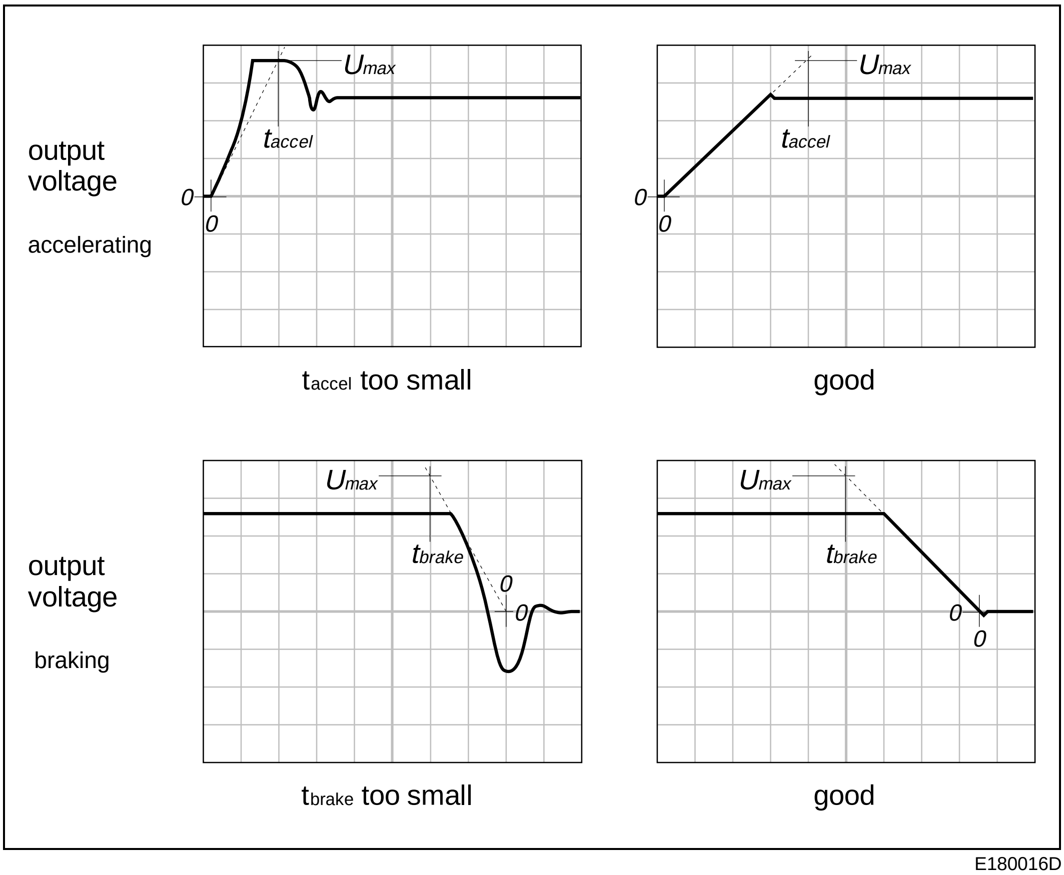
5.6.3 Accelerating/braking times

The values for t_{accel+}, t_{accel-}, t_{brake+} and t_{brake-} (storage locations 3/**35**...**38**) have already been programmed (section 5.3). Here you should only check whether the settings are correct.



E180011E

Too <u>short</u> times negatively influence the control response and may possibly result in inadmissible mechanical and electrical loads on the drive system. The oscillograms below show the behaviour of the output voltage if the times are set too small and in case they are set properly.



By means of a current monitor at the drive amplifier you may find out whether the current limit is reached during the process of acceleration or braking. In the affirmative, the time programmed is too short.

Another possibility to find out whether the values are too small is observing the contouring distance Delta_s: major fluctuations during accelerating or braking indicate that the appropriate values are too small.

Times being too <u>long</u> are uncritical for the control but do, however, negatively influence the duration of positioning. The Controller limits the values to a maximum size, which, however, can only be attained under most unfavourable conditions (see storage location 3/**35** in appendix A).

✔ Check by manually displacing the drive into both directions at low speed ("**>**" and "**<**") in accordance with the information supplied above, whether the times are too short or whether they could be reduced (higher mechanical load!).

✔ If necessary, program other values (storage locations 3/**35**...**38**) and check again the drive behaviour with these new values.

5.6.4 Jerk time

Slight overshoots at the end of the accelerating and braking period (see oscillogram above) can be minimized by programming a jerk time t_{jerk}, this, however, at the expense of a slightly increased duration of positioning. By means of the jerk time the positioning characteristic of the drive is fixed (refer to section 4.4.2).

✔ Storage location 3/**39**: jerk (time) t_{jerk}

Set the value by manually displacing at low speed rate (working speed) in such a way, that the drive shows the desired behaviour for the later positioning procedures.

Note: Because of internal calculation reasons the jerk time is limited to a maximum value (see storage location 3/**39** in appendix A).

Now the control parameters are properly set or optimized for the drive. Finally, further machine parameters must be set to configure the Controller for the installation.

5.7 Controller configuration

Since the programming of further machine parameters is rather individual and depending on the application, we cannot provide any concrete procedure nor can we give any values.

We recommend you to take the survey at the beginning of appendix A and the following table to decide on further parameters to be determined.

You may, however, browse through the machine parameters at the Controller setting the required functions and values. Please do also observe the information on the individual storage locations in appendix A (and possibly appendix O).

Here are some basic points:

✔ Determine, among others, for each unit of which elements a nominal value sentence should consist and whether the axes should be operated in path control: storage locations 2/**1**...**5**; confirm the security inquiry `Delete unit?` by pressing the ⊛ key.

✔ Determine, among others, the following for the axes parameters:

- Storage locations 3/**9**...**17**: calibration functions and values, if incremental encoders are used.
- Storage location 3/**20**: adaptation of the keys and/or signals for the manual drive control to the installation-specific directions
- Storage locations 3/**21**...**24**: absolute speed values for the manual drive control
- Storage locations 3/**27** and 3/**28**: switch-off points $S_{dead+/-}$ for control stabilization in the nominal or inoperative position
- Storage locations 3/**34**: final working speed rate for positioning procedures without nominal value setting in the sentence
- Storage locations 3/**40** and 3/**41**: range of tolerance $Tol_{+/-}$ for the nominal position, where the *actual=nominal* signal is output
- Storage locations 3/**42** and 3/**43**: admissible contouring distance $S_{max+/-}$; if this value is exceeded the positioning process will be interrupted; this value should be derived from the largest operational contouring distance (Delta_s), including a generous increased factor of safety ($S_{max+/-} \approx 10 *$ Delta_s)
- Storage locations 3/**45** and 3/**46**: adaptation to the requested input format for nominal speed values in the sentence
- Storage locations 3/**47**...**50**: control functions for the various modes and states of operation

- Storage locations 3/**56**...**60**: functions and values for a parking position
- Storage locations 3/**61**...**70**: functions and values for range signals
- Storage locations 3/**71**...**74**: input control/monitoring and limit switch functions
- ...

Now (first) commissioning is completed.

6 Trouble shooting

6.1 Warning and error messages

Apart from various operating state messages, the Controller outputs an appropriate warning or error message in certain operating situations in displays B and C. Each message must be acknowledged by pressing any key. For warning messages requiring a decision, the  key means confirmation whereas any other key aborts the respective function.

The following overview in alphabetic order shows the various error messages (and the order in which they are dealt with). The descriptions do not represent the whole range of possibilities. Messages valid only for certain **options** are dealt with at their corresponding descriptions (refer to appendix O).

Messages:

```
Delete memory ?
Delete program?
Delete unit ?
EEPROM error
Fatal error 1
Inactive auto cal
Inactive entry
Invalid auto cal
Invalid delete
Invalid entry
Invalid pieces
Invalid prog.end
Invalid program
Invalid sentence
Mem.card error
Mem.card missed
Memory overflow
Only for s.begin
Only in reset
Over 999 sentence
Param. error
Parkpos. too high
Parkpos. too low
Position too high
Position too low
Prog.end exists
Prog.end missed
read m.card ?
Referenc too high
Referenc too low
Speed too high
write m.card ?
Write protect
```

Warning or error message	Cause, situation	Remedy, reaction
`Delete memory ?`	The [F3]+[7] keys resp. were pressed directly after entering the machine parameter programming mode (all machine parameters will be reset to 0 and all storage locations for the nominal values will be cleared)	Confirm the safety inquiry with or abort function with any other key
`Delete program?`	After the selection of a program the [F3]+[7] keys resp. have been pressed (instead of entering the sentence number)	Confirm safety inquiry with or abort function with any other key
`Delete unit ?`	• In the nominal value programming mode, the [F3]+[7] keys resp. have been pressed after selection of the unit (instead of the program number input);	Confirm the safety inquiry with or abort function with any other key
	• In the machine parameter programming mode, either – the unit/axis assignment (storage locations 1/**3** to 1/**5**) or – the configuration of the nominal value sentence for the unit (storage locations 2/**1** to 2/**5**) has been changed (all storage locations, i.e., programs of the respective unit are cleared)	As before

Warning or error message	Cause, situation	Remedy, reaction
`EEPROM error`	Non-correctable error in the memory used by the power failure security (see also the info at the beginning of chapter 5) (can only occur during nominal value entry or when reading a memory card)	Replace the Controller
`Fatal error 1`	Non-correctable RAM error (can only occur if a defect Controller is switched on; the message cannot be cleared)	Replace the Controller
`Inactive auto cal` *automatic calibration function is inactive*	You tried to start an automatic reference search routine ([F1]+[2] resp. [⊕]) although this function has not been enabled in the machine parameters for the respective axis	Enable this function at the corresponding axis parameter (storage location 3/**9**)
`Inactive entry`	In the Automatic mode, you tried – to select a program ([F3]+[1]) or – to enter a reference value, a correction value or a parking position directly ([F1]+[1]/[3]/[4] resp. [⊕]/[↦]/–) although this function has not been enabled in the machine parameters for the respective unit/axis	Activate program selection or direct input possibilities for the respective unit parameters (storage location 2/**6**) or axis parameters (storage location 3/**7**/**9**/**57**)

Warning or error message	Cause, situation	Remedy, reaction
`Invalid auto cal` *starting the automatic calibration is invalid*	You tried to start an automatic reference search routine (F1 + 2 resp. [key]) although the unit of the corresponding axis has not been reset	Reset the Controller for the corresponding unit
`Invalid delete`	You tried to delete – the only nominal value sentence of a program or – the end of the program itself (F3 + [key] resp. [key])	Acknowledge with any key
`Invalid entry`	• In the Automatic mode you tried – to change the program (F3 + 1), or – the direct entry of a reference value, a correction value, or a parking position was activated (F1 + 1 / 3 / 4) although the unit of the respective axis has not been reset	Reset the Controller for the respective unit
	• In the nominal value programming mode, you have entered an invalid I/O number for an IF command	Enter a valid I/O number (refer to section 4.14)
`Invalid pieces` *invalid number of pieces*	0 was entered for the nominal number of pieces	Enter the correct piece number

Warning or error message	Cause, situation	Remedy, reaction
Invalid prog.end *invalid end of program*	• You tried to define the end of program for a program that is still empty ((F3)+(9) resp. (Def. Prog. End))	First, enter nominal values or exit the program
	• You tried to insert a sentence when the end of program is displayed ((F3)+(➔) resp. (☰))	Scroll back to the sentence after which a new sentence is to be inserted
Invalid program	• When copying sentences ((F3)+(8) resp. (Copy)), a non-existent program has been specified as source	Enter a correct program number
	• When powering on the device: – you had switched off while the message Saving program has been issued, or – the device is defect	Clear all nominal value storage locations (i.e., carry out the clearing process for each programmed unit); then switch the device off and on again. If the message is still indicated, the device must be replaced. In the other case, reprogram the nominal values. The Controller cannot be started if the message has been acknowledged without carrying out the above step(s)

Warning or error message	Cause, situation	Remedy, reaction
Invalid sentence	• When copying sentences ((F3)+(8) resp. (Copy)) a non-existent sentence has been specified as source	Enter the correct sentence number
	• When copying in the over-write mode (destination sentence number has been confirmed with (⇨) only), the number of sentences within the actual program is lower than that of the sentences to be copied	Reduce the number of the last sentence to be copied or use the insert mode (confirm the sentence number at "to" with (F3)+(⇨) resp. (⊟))
Mem.card error	• Memory card used is defect or has been used so far with another Controller type (GEL 8310 ↔ GEL 8610)	Use another card
	• You tried to read a card which has not been written yet (check sum error)	Exchange the card or select ‘write’ instead of ‘read’
	• You tried to read all parameters although the axis parameters have not been stored on the card in all the previous writing processes	Modify the storage location 1/**7** (GEL 8610: 1/**10**) accordingly
	• The memory card has been removed while writing or reading	Insert the card again and repeat writing/reading
Mem.card missed *missing memory card*	• The memory card has not been inserted correctly	insert the memory card correctly
	• A card non-formatted by the factory is used	use another card
	(occurs when trying to read or write a memory card)	

Warning or error message	Cause, situation	Remedy, reaction
`Memory overflow`	All nominal value storage locations are already used. Occurs – after entering the last possible nominal value when trying to insert another sentence or to copy several sentences in the insert mode – after trying to create a new program if there is no space left for at least two sentences	Define the end of program in the actual program ([F3]+[9] resp. [Def. Prog. End]); if required, delete sentences in another program in order to free-up new memory space
`Only for s. begin` *at the beginning of the sentence only*	For a sentence already begun, i.e., starting with the second or a further nominal value in the actual sentence, you tried – to define the end of program ([F3]+[9] resp. [Def. Prog. End]), or – to set a program flow instruction at this position ([F3]+[1] ... [F3]+[4])	First, enter the remaining nominal values of the actual sentence
`Only in reset` *in reset state only*	You tried – to copy sentences ([F3]+[8] resp. [Copy]) or – to delete ([F3]+[⫽] resp. [⫽]) or to insert a sentence ([F3]+[⇨] resp. [▤]) – to read data from the memory card ([F3]+[⇨] resp. [⇄]) although there is still at least 1 unit in the started or interrupted state	Reset the Controller for <u>all</u> units

Warning or error message	Cause, situation	Remedy, reaction
Over 999 sentence	• 998 sentences have already been programmed (the end of program does also count as a sentence) and you now tried to enter another nominal value or to insert a sentence	Terminate the program
	• When copying in the <u>insert</u> mode, too many sentences have been selected so that the sum of the sentences existing and still to be copied would exceed the value 998	Reduce the number of sentences to be copied
Param. error *parameter error*	An inadmissible value has been entered for a machine parameter (occurs when trying to exit the programming mode of machine parameters)	Confirm the message with ; the invalid parameter is activated and can now be corrected If the message is acknowledged with (e.g.) ESC, i.e. ignored, no unit can be started and the programming mode of nominal values cannot be activated. In such a case, change an arbitrary parameter, save it, and exit the programming mode. Then, the invalid parameter will be indicated again.
Parkpos. too high *parking position is too high*	Input monitoring: value of parking position exceeds the specified »Pos. max« value (storage location 3/**72**)	Enter a lower value or specify »Pos. max« new
Parkpos. too low *parking position is too low*	Input monitoring: value of parking position is lower than the specified »Pos. min« value (storage location 3/**71**)	Enter a higher value or specify »Pos. min« new

Warning or error message	Cause, situation	Remedy, reaction
`Position too high`	Input monitoring: value of the nominal position/ length exceeds the specified »Pos. max« value (storage location 3/**72**)	Enter a lower value or specify »Pos. max« new
`Position too low`	Input monitoring: value of the nominal position/ length is lower than the specified »Pos. min« value (storage location 3/**71**)	Enter a higher value or specify »Pos. min« new
`Prog. end exists` *end of program already exists*	You tried to redefine the end of program for an already established program ([F3]+[9] resp. [Def. Prog. End])	Acknowledge the message with any key
`Prog. end missed` *missing end of program*	You tried to exit a still empty or newly created program ([ESC] or [F2]+[0] resp. [Auto])	• Entries have already been made: do <u>not</u> confirm the message with [⇨] (the program is not exited) and define the end of program ([F3]+[9] resp. [Def. Prog. End]) • An empty program has been selected (by mistake): confirm message with [⇨] (the program is exited)
`read m.card ?` *read memory card?*	The [F3]+[⇨] keys resp. [memory card key] have been actuated in order to overwrite the machine parameters and nominal values in the EEPROM of the Controller with the values from the memory card	Confirm with [⇨] or abort with any other key

Warning or error message	Cause, situation	Remedy, reaction
`Referenc too high` *reference measure is too high*	Input monitoring: value of the reference measure exceeds the specified »Pos. max« value (storage location 3/**72**)	Enter a lower value or specify »Pos. max« new
`Referenc too low` *reference measure is too*	Input monitoring: value of the reference measure is lower than the specified »Pos. min« value (storage location 3/**71**)	Enter a higher value or specify »Pos. min« new
`Speed too high`	A nominal speed value has been entered that is larger than the programmed maximum value (storage location 3/**32**) considering a programmed multiplier and a decimal point for the speed rate input (storage locations 3/**45** and 3/**46**)	Enter a lower value
`write m.card ?` *write memory card?*	The [F2]+[→] keys resp. [key] have been actuated in order to safe the machine parameters and nominal values on the memory card inserted	Confirm with [→] or abort with any other key
`Write protect`	You tried to write on a memory card although the write protection has been activated at the machine parameter 1/**6** (GEL 8610: 1/**9**)	Modify the machine parameter accordingly

If **display B** shows a **point** only, the supply voltage remains under the **minimum** value stated in appendix C under ‘Operational data’. In this case the Controller stops working, because it is in an **undefined operating state**, which has to be terminated as quickly as possible: Switch off the equipment and perform the necessary measures, so that the supply voltage will remain stable inside the limits.

6.2 Fault memory

Up to **20** faults that appeared during the operation are stored successively and non-volatile in the memory of the Controller. In addition, each appearing fault pushes the one at the bottom out of the memory.

Fault messages can be accessed in every operating state of the Automatic operation mode after pressing [F1]+[0] resp. [Error] (GEL 8610).

The finally stored fault is then displayed. The `no error` message is output if no fault has appeared yet.

Representation format (example)**:**

- Display A:

	u.	*c*	*c.*	*t*	*t.*	*n*	*n*

u : unit or axis which caused the fault (1 ... 3 / 6);
0 = system fault

c c : current number of indicated fault (1 ... 20)

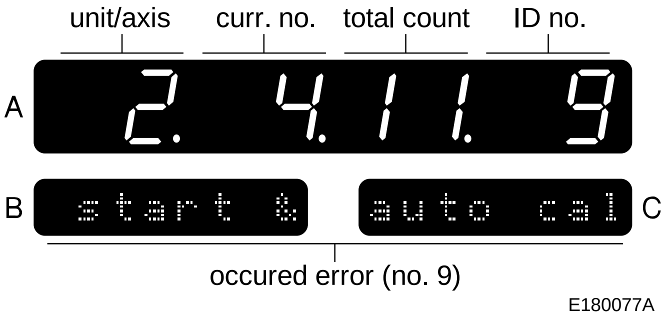
t t : total count of faults stored (1 ... 20)

n n: identification no. of the fault (see table)

- Display B/C: plain text of the fault (see table)

  If the Controller is equipped with a serial interface (module V) **with clock**, the date and time of the occurrence of the fault can be displayed: [·] resp. [clock].

Example: A reference search routine is released for an axis of the 2nd unit although the unit has been already started; as this (intended) action shows no effect, a corresponding fault message is stored (which is the fourth message stored; at the time of the fault polling, the number of faults stored totals 11).



The [↓]/[↑] keys can be used to scroll through the messages if several fault numbers are present in memory.

If the key is pressed when displaying the **last** fault, **all** fault numbers can be removed from memory if the safety inquiry `Delete all ?` is confirmed with . With ESC the action is aborted. The displays are then switched to the normal operating status again, i.e., the inquiry function is exited.

An **individual** fault message can be deleted when it is displayed by pressing and confirming the `Delete this ?` safety inquiry with , too. With ESC the action is aborted. The number of faults and the running number of the faults above the deleted are then reduced by 1. If both were previously 1 then the displays are switched to the normal operating state again, i.e., the inquiry function is exited.

The inquiry function can be exited at any time using ESC.

Table of faults:

No.	Displays B/C	Unit	Axis	Description
1	`= Pos. > Pos. max`		X	Actual position exceeds the »Pos. max« software limit switch (storage location 3/**72**); refer to section 4.11.1
2	`= Pos. < Pos. min`		X	Actual position is less than the »Pos. min« software limit switch (storage location 3/**71**); refer to section 4.11.1
3	`! Pos. > Pos. max`		X	Nominal position exceeds the programmed »Pos. max« maximum value (storage location 3/**72**), refer to section 4.11.1; occurs if »Pos. max« is changed after specifying the nominal value or when processing lengths in the absolute dimensions system
4	`! Pos. < Pos. min`		X	Nominal position is less than the programmed »Pos. min« minimum value (storage location 3/**71**), refer to section 4.11.1; occurs if »Pos. min« is changed after specifying the nominal value or when processing lengths in the absolute dimensions system
5	`Delta_s> S max +`		X	Absolute value of the positive control deviation (contouring distance) is larger than the programmed »S max +« maximum value; refer to storage location 3/**42**

No.	Displays B/C	Unit	Axis	Description
6	Delta_s< S max -		X	Absolute value of the negative control deviation (contouring distance) is larger than the programmed »S max -« maximum value; refer to storage location 3/**43**
7	stop & auto cal		X	*Search for reference* signal was preset with Low level at the *stop* input or the unit being in the interrupted state
8	reset & auto cal		X	*Search for reference* signal was preset with High level at the *reset* input
9	start & auto cal		X	*Search for reference* signal was preset with the unit being in the started state
10	m.Drive& auto cal		X	*Search for reference* signal was preset during manual positioning of the drive
11	ProgPar& auto cal		X	*Search for reference* signal was preset during the programming of machine parameters
12	MemEdit& auto cal		X	*Search for reference* signal was preset during the recalculation of machine parameters, i.e., directly after terminating the programming operation
13	stop & start	X		*Start* signal was preset with Low level at the *stop* input
14	reset & start	X		*Start* signal was preset with High level at the *reset* input
15	autoCal& start	X		*Start* signal was preset although at least one axis of the unit performs an automatic reference search routine
16	m.Drive& start	X		*Start* signal was preset during manual positioning of a drive of the unit
17	ProgPar& start	X		*Start* signal was preset during the programming of machine parameters
18	MemEdit& start	X		*Start* signal was preset during a program structure change (inserting or deleting of nominal value sentences/programs and serial transmission of already existing programs and of machine parameters)

No.	Displays B/C	Unit	Axis	Description
19	start ➡ calib.	X		*Start* signal was preset although the reference measure was not set for all axes to be calibrated; refer to storage location 1/**2**
20	invalid program	X		*Start* signal was preset although no valid program was selected
21	start & Prog.Par	X		The started operating state was aborted by the programming of a machine parameter
22	autoCal& Prog.Par		X	An autom. reference search routine was interrupted by the programming of a machine parameter
23	m.Drive& Prog.Par		X	Manual positioning was aborted by the programming of a machine parameter
24	autoCal& MemEdit		X	An autom. reference search routine was aborted by the modification and recalculation of machine parameters
25	m.Drive & MemEdit		X	Manual positioning was aborted by the modification or recalculation of a machine parameter
26	framing error			Stop bit error at the serial interface (wrong polarity: Low level)
27	ser.com. error			Error during serial transmission (parity, overrun, check)
28	axis in HW-max		X	Axis triggered the upper hardware limit switch (MAX) (refer to section 4.11.2)
29	axis in HW-min		X	Axis triggered the lower hardware limit switch (MIN) (refer to section 4.11.2)
30	invalid sentence	X		A non-existing sentence was selected via the data input or by the ‘JMP sent’ instruction
31	more as 5 JUMPs	X		More than 5 consecutive jump instructions are specified in the program (‘JUMP Pr.’ and/or ‘JMP sent’, refer to section 4.14)
32	too many CALLs	X		More than 20 nested sub-routine calls (‘CALL Pr.’) are specified in the program (refer to section 4.14)

No.	Displays B/C	Unit	Axis	Description
33	IF from reset	X		After a start from a reset state, an IF instruction (‘IF I/O’) was executed by the program without performing a previous positioning (refer to section 4.14)
36	Watchdog Reset			Controller reset, caused by external noise (EMC measurements!) or an internal fault (repair is possibly required)

Tip

Please evaluate the fault memory if the drive(s) of an unit **cannot be started.** Surely, you will find there the cause for the problem.

Exceptions (no fault message):

- A parameter error has not been removed (refer to section 6.1, Param. error)
- Axes which are still not associated to a unit (system parameters) can, of course, also not be started

The following causes (no fault message!) can be the reason why a drive **cannot be positioned manually** although the corresponding storage locations are correctly programmed:

- no High level is present at the *stop* input of the corresponding unit
- High level is present at the *reset* input of the corresponding unit
- the unit of the corresponding axis is in the started state
- the unit of the corresponding axis is in the interrupted state, manual positioning was allowed, however, for the reset state only (storage location 3/**18**)
- machine parameters are just recalculated after a change or serial transmission

7 Mounting instructions

7.1 Cable connections

The compact design of the Controller and the variety of modules and connection properties has consequently lead to the use of relatively small dimensioned terminal strips. For a **good electrical contact** and **mechanical grip** of the cables in the terminal strips as well as for a **safe insulation** we strongly recommend that stranded cables be fitted with wire end ferrules to DIN 46228 part 4, which are fixed permanently using a special crimping tool. If two or more thin cables are to be connected to one terminal, the use of twin wire end ferrules will be advantageous.

For tightening the terminal screws use a screw driver with the following blade size:

- 0.4×2.5 mm for terminal strips with a 3.8 mm grid (screw M2) as C, D, P, W, Y, and Z
- 0.5×3.5 mm for terminal strips with 5.1 mm and 7.6 mm grids (screw M3) as N and R

7.2 EMC measures

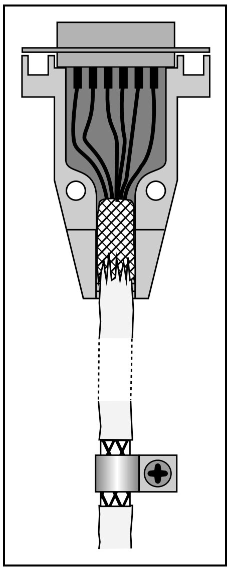
In order to achieve highest electromagnetic compatibility (EMC) or to maintain EMC certified for the devices with the CE marker pay special attention to the screening and earthing. In general:

➨ Earthing connections should be kept **as short as possible** and with a **large cross-section** (e.g. low-inductance braided cable, flat-band conductor)

➨ Mounting plates as well as the control cabinet must be well earthed

➨ All un-screened cables should be kept **as short as possible**

➨ Connect screens at **both cable ends** with **as large a surface area** as possible

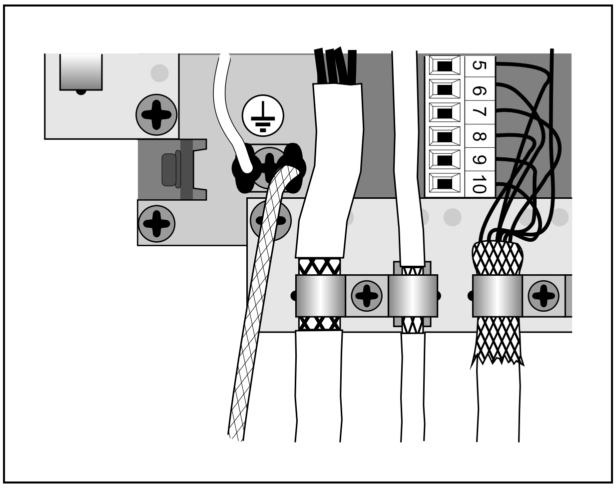


X035AH

➨ Only connectors with **metal housings** or a housing made of metallized plastic should be used and the screen connected directly to the strain relief of the connector with as large a surface area as possible

If the connector does not have special strain relief clamp, it is advised to provide adequate clamping between the two halves of the housing. If necessary widen the cable with shrink-sleeve before folding over the screen.

The same applies to the cables at the terminal strips for the CE devices:



X180014J

➨ Signal and control lines should be laid **separated** from the power cables

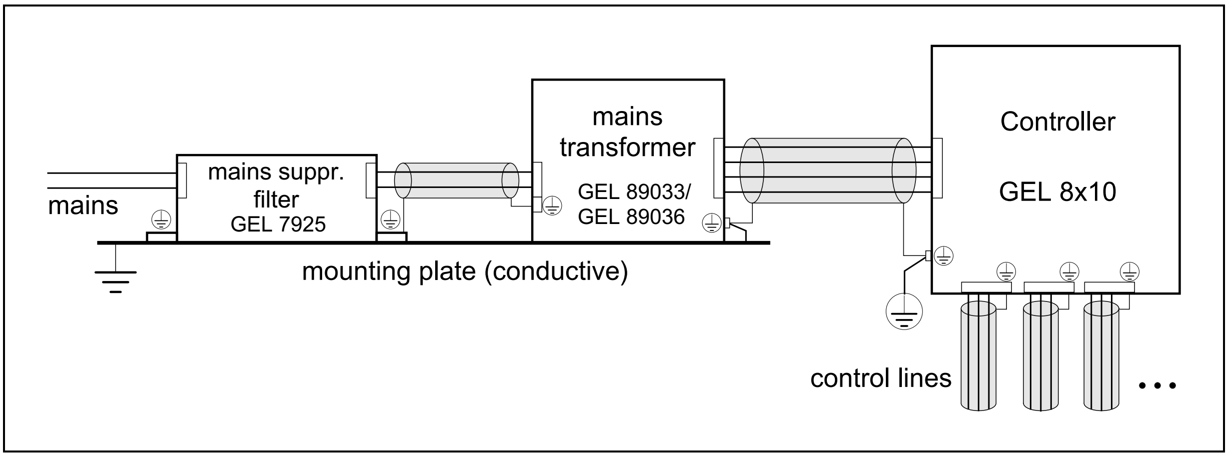
➨ Should **potential differences** exist or occur between machine- and electronic-earth connections, measures should be taken so that **no compensating currents** can flow through the screen (e.g. potential equalisation line with large cross-section or cable with double screen, where only one end of each screen is to be connected at different ends)

➨ If **inductive loads** (relays, contactors) are connected to the logic outputs, measures for **spark suppression** should be provided (recovering diode or RC network parallel connected and close to the coil)

For the power supply of the Controllers **without CE certification** the use of the mains transformer – offered as accessory, cf. appendix C – is **strictly** recommended.

By use of a mains suppression filter – an accessory, too – you can improve the EMC behaviour in an unfavourable environment.

To this configuration the following screening and earthing principle is applicable (when using another source of current equivalent measures are to be performed):



a) Mains suppression filter

To achieve full functionality of the filter you must fix it to a bright and well earthed mounting plate, paying attention to having good contact at both sides. The screen (earth) is to be connected to the special labelled screw terminal on the filter.

b) Mains transformer

The transformer is to be fixed to the mounting plate, too. Contacting has to be performed with as large a surface area as possible (remove a possibly existing coat of lacquer). Additionally, earth the transformer via a short cable (≥ 2.5 mm^2) to the mounting plate using the twin blade terminal (6.3 mm) on the transformer.

Connect the mains earth wire resp. the screen of the mains cable coming from the filter to the ‘PE’ labelled terminal.

Connect the screen of the Controller supply to the twin blade terminal on the transformer.

c) Controller

Each inserted module is earthed via a special contact socket in the upper metal angle (metal frame with CE devices) at the back of the Controller.

Connect
– the screen of the transformer cable (for devices without CE certification) and
– the earth cable coming from the switch cabinet (door)
to the twin blade connector on the metal angle/frame.

Screen earthing is performed
– by connecting the screens to the respective terminal strips or plugs of the individual modules (devices without CE certification, see appendix B) resp.
– via the cable clips on the earthing angles screwed down on the metal frame (devices with CE certification, see figure on page 7-2).

7.3 Replacing modules

Before any service work is carried out at the Controller, make sure that the **whole** Controller, i.e. including the inserted module, is dead. This applies, in particular, to the **relay cards R** used (see Fig. 1): operating voltages up to **240 VAC** are possible!

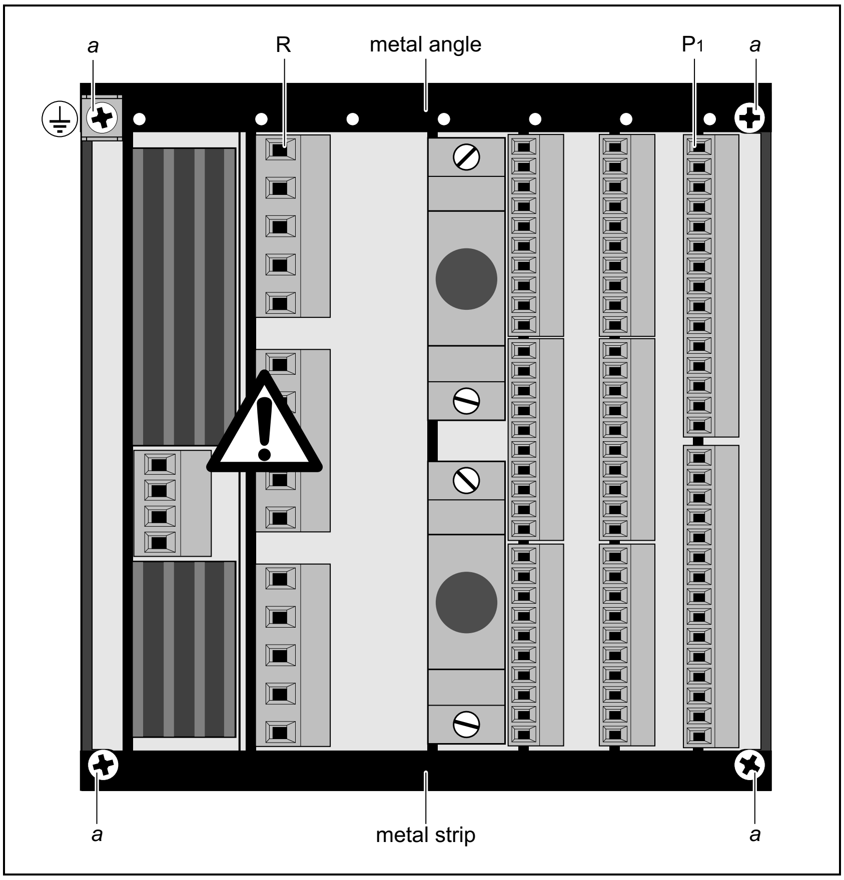
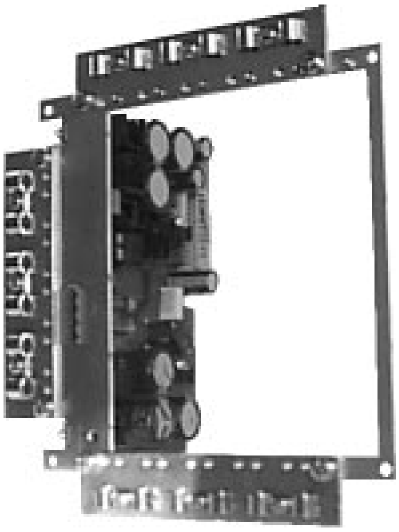


Fig. 1 (here: Controller GEL 8310 without CE certification) E183083A

7.3.1 Disassembly

✔ Remove optional memory card, if inserted.

devices **with** CE certification	devices **without** CE certification
✔ Unplug all connectors from the modules	✔ Disconnect earth wire from the metal angle (blade connector 6.3 mm)
✔ Remove 4 recessed head screws at the corners of the metal frame (with the GEL 8610 there are 2 additional screws in the middle top and bottom)	✔ Remove metal angle and strip: unscrew 4 recessed head screws (Fig. 1, *a*); with the GEL 8610 there are 2 additional screws in the middle top and bottom

devices **with** CE certification	devices **without** CE certification
 ✔ Pull out the metal frame with mounted power supply card ✔ Pull out the desired module	✔ Pull out the desired module preferably grasping one of the (still) plugged terminal strips

Due to the compact construction, it may be possible that a card is jamming slightly. If this is the case, press the card next to it slightly aside or also pull it out (partially).

✔ Unplug the terminal strips still being plugged on the module

All terminal strips are coded and clearly labelled (see appendix B), so that they cannot be confused later on.

7.3.2 Assembly

✔ Insert the module in the guiding rails and push it in

The Controller housing contains each 2 adjacent guiding rails. Both power and P1 modules (see Fig. 1) are to be inserted into the left guiding rails. All other modules are to be inserted into the right rails (see also appendix B under ‘Module arrangement …’).

If a module strucks or blocks during insertion press the module next to it slightly aside or pull it out partially so that both modules can be pushed in together.

Special notice for module P1 with optional memory card:

Due to the guide frame for the memory card mounted on the front panel, the module might abut against this frame. In this case press the card slightly to the left (using a flat tool) so that it can pass the frame on the left.

devices **with** CE certification	devices **without** CE certification
✔ Place the metal frame with mounted power supply card	✔ Plug all terminal strips (identical designations on terminal strip and socket!) ✔ Place the metal angle

Mind that all earth pins of the modules fit correctly in the associated sockets of the metal frame/angle.

devices **with** CE certification	devices **without** CE certification
✔ Screw the metal frame ✔ Plug all connectors and terminal strips	✔ Screw the metal angle and strip including the twin blade terminal for earthing ✔ Connect earth wire to the metal angle Mind that the cable lug does not contact the screw of the heat sink. In this case (internal) ground would be connected to earth.

7.4 Replacing an EPROM

The following statements are related to the program EPROM. The equivalent is valid for EPROMs on other modules (e. g . SSI interface and PROFIBUS).

The program EPROM is located

– on the main board behind the keyboard of the Controller for devices **with** CE certification: **PLCC** type resp.
– on the P1 module (control inputs/outputs) for devices **without** CE certification: **DIL** type.

7.4.1 PLCC EPROM

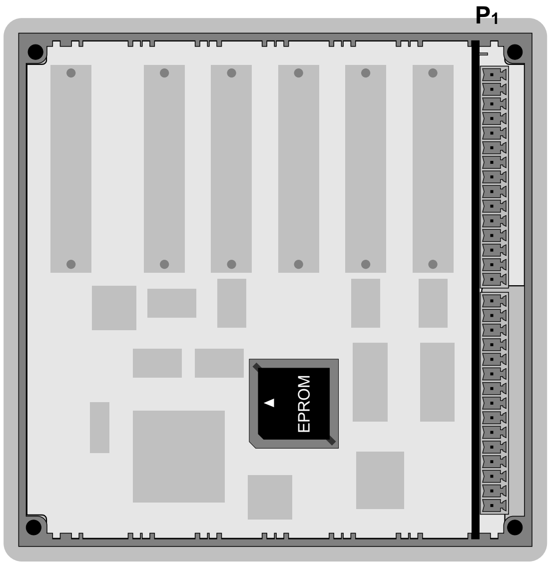
For the replacement all modules and blind plates have to be removed from the Controller (GEL 8610: on the right half only), the P module may be excepted from this with built-in memory card option.

✔ Disconnect all current sources from the Controller

✔ Remove modules as described in section 7.3.1

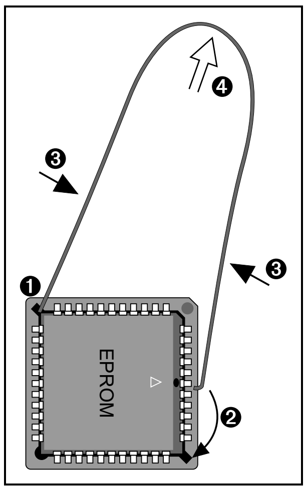
The EPROM is accessible now.

It will advantageously be removed from the socket by means of a special extractor tool. Starting from the fact that this (expensive) tool is not available in most cases, a U-shaped wire is enclosed to the chip as a vehicle to pull it out of the socket:

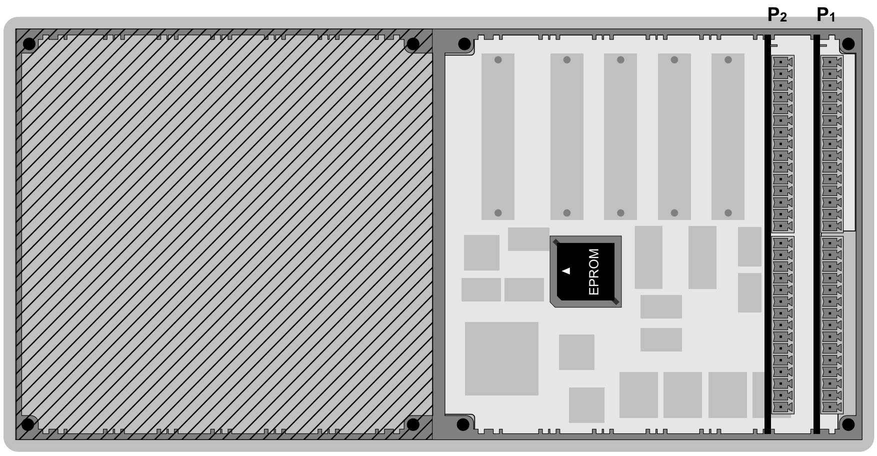


GEL 8310 X183003C

✔ Bring the bended ends of the bow under the chip (❶, ❷), press the bow slightly (❸), and pull it up (❹)



X061AH

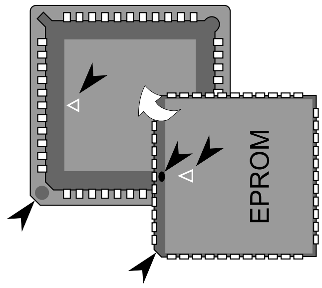


GEL 8610

X186003C

When **inserting** the chip notice the **correct orientation** with respect to the socket. For this, both parts have special marks (illustrated in the figure below by black arrows):

- a skewed corner
- a triangle (printed on the chip, relief in the socket)
- a round indentation in the bevel of the memory chip



X060AH

ATTENTION

A **wrongly inserted EPROM** can cause **hardware defects** that may not be sensible. Therefore, check the correct orientation of the chip before assembling the device.

✔ Insert modules and assemble the Controller as described in section 7.3.2

7.4.2 DIL-EPROM

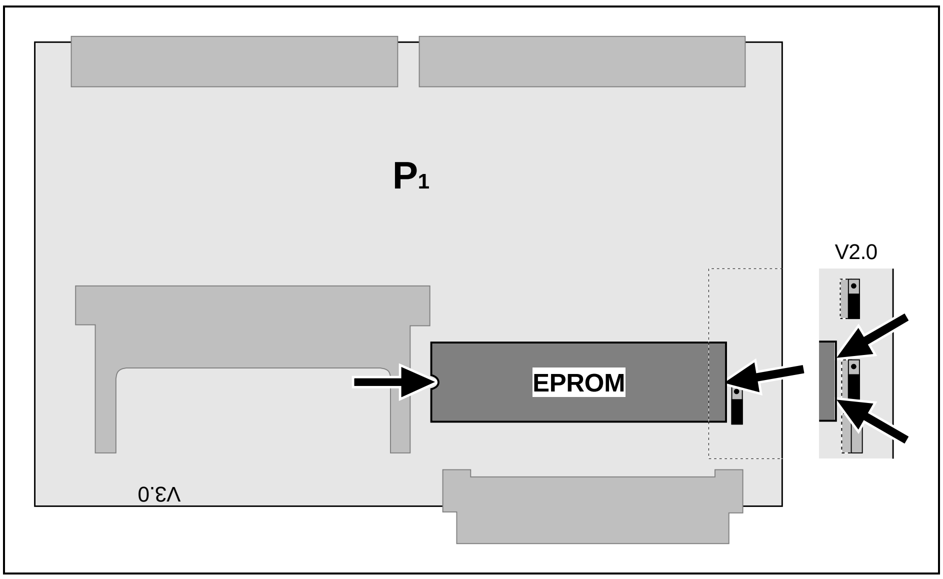


Fig. 2

X180083B

✔ Remove the module P1 as described in section 7.3.1

✔ Lever the EPROM alternately at both sides as illustrated in Fig. 2

Note for V2.0 modules:

The two jumpers and the capacitor on the right side of the EPROM (Fig. 2) must not be bent back into the vertical position as otherwise the module cannot be pushed in.

✔ Check if the pins of the new EPROM show the correct angle for the socket. If not, bend the pins to the required measure according to Fig. 3, parts *1* and *2*

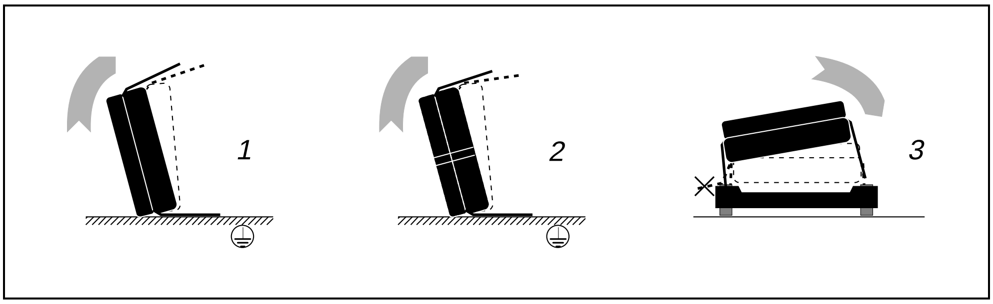


Fig. 3 X180083C

✔ Insert the EPROM in the socket considering the correct position (mark on the left side, cf. Fig. 2): Fig. 3, part *3*

Please make sure that all pins fit correctly in the contact bushes.

✔ Insert the module and assemble the Controller as described in section 7.3.2

Overview of the storage locations

1. **System** parameters (1st level)

1:	Language	
2:	Power failure security	
3:	Configuration unit 1	
4:	Configuration unit 2	
5:	Configuration unit 3	
/ 6:	Configuration unit 4	*GEL 8610 only*
/ 7:	Configuration unit 5	*GEL 8610 only*
/ 8:	Configuration unit 6	*GEL 8610 only*
6 / 9:	Memory card	*GEL 8310 / 8610*
7 / 10:	Special function with memory card	*GEL 8310 / 8610*
8 / 11:	Password inquiry for nominal value programming	*GEL 8310 / 8610*
9 / 12:	Specification of the password	*GEL 8310 / 8610*
10 / 13:	Keyboard lock (terminal P21)	*GEL 8310 / 8610*
11 / 14:	Serial interface	*GEL 8310 / 8610*
12 / 15:	Device number	*GEL 8310 / 8610*
13 / 16:	Protocol	*GEL 8310 / 8610*
14 / 17:	Service (deactivation of passwords)	*GEL 8310 / 8610*
15 / 18: ⋮ 21 / 24:	— *(reserved)*	

2. **Unit** parameters (2nd level)

1: Number of pieces/auto start per sentence
2: Machine functions
3: Speed rate
4: Identification text
5: Path control
6: Program selection
7: Sentence and program selection
8: Sentence and program number to data output
9: Program processing signals to data output
10: Auto start
11: Mode for signal output
12: Program processing signal 'Reset'
13: — *(reserved)*
14: — *(reserved)*

3. **Axis** parameters (3rd level)

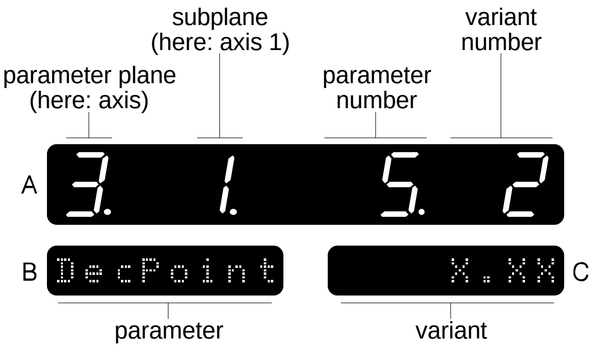
1: Actual value adjustment
2: Count direction
3: Multiplier
4: Multiplier for actual value display
5: Decimal point
6: Correction value
7: Direct entry of correction value
8: Rotary table

9: Manual calibration functions
10: Setting of reference measure
11: Automatic reference search routine
12: *Reference fine* signal
13: *Reference coarse* signal
14: *Reversing switch* signal
15: First reference measure
16: Reference positioning speed rate
17: Reversing speed
18: Operating status for manual positioning
19: Manual positioning
20: Polarity for manual positioning control
21: Slow speed rate, forward
22: Fast speed rate, forward
23: Slow speed rate, reverse
24: Fast speed rate, reverse
25: Polarity of the analogue output
26: Voltage range of the analogue output
27: Positive dead range
28: Negative dead range
29: Minimum positive voltage
30: Minimum negative voltage
31: Maximum voltage
32: Maximum speed rate
33: Control factor
34: Operating speed rate
35: Maximum acceleration, forward
36: Maximum acceleration, reverse
37: Maximum braking, forward
38: Maximum braking, reverse
39: Jerking time
40: Positive tolerance
41: Negative tolerance
42: Maximum positive contouring error
43: Maximum negative contouring error
44: Measurement system
45: Multiplier for speed rate values
46: Decimal point for speed rate values
47: Position control in the interrupted/reset state
48: Control conditions after reaching the nominal position
49: Control of automatic calibration
50: Control of manual positioning
51: Time for opening the brake
52: Time for closing the brake
53: Zero point adjustment for angle encoders
54: Resolution of absolute encoders
55: Enable signal for absolute encoders
56: Parking function
57: Direct entry of a park position
58: Value of park position

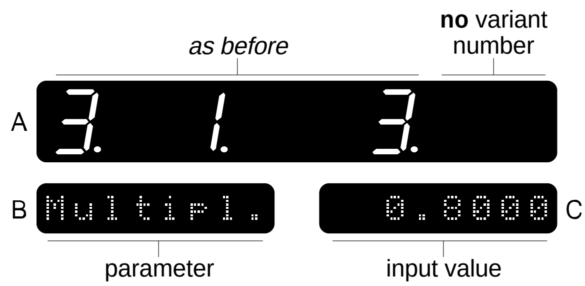
59: Parking speed rate
60: Machine functions for parking
61: Output of range signals
62: Function for range signals
63:
⋮ } Start and end values of the R1 to R4 ranges
70:
71: Minimum position value
72: Maximum position value
73: Software limit switches
74: Hardware limit switches
75: External data input of a nominal position/length
76: External data input of a correction value
77: External data input of a speed rate
78: — *(reserved)*
79: — *(reserved)*
80: Data output of nominal positions
81: Data output of actual positions
82: Data output of correction values
83: — *(reserved)*
84: — *(reserved)*
85: Value of the second reference measure
86:
⋮ } — *(reserved)*
89:
90: Zero Delta_s
91: Calibration after powering-on

Parameter format

a) Parameter with variant selection



b) Parameter with value input



For more details refer to section 3.3. The programming mode is activated by pressing [F2]+[1] and entering the password ('9228'). It is terminated via [F2]+[0] resp. [Auto] or [ESC].

Explanations on the representation used

Example:

❶ 1	❷ **1**	❸ **Language**	❹ Languages
Determine the language for the display of texts ❺			
0		german	texts are displayed in German
1		english	texts are displayed in English
❻		❼	❽

❶ Number of parameter level (from display A):
1 = system parameters, 2 = unit parameters, 3 = axis parameters

❷ Number of parameter storage location within the level (from display A), scrolling with [⬇]/[⬆]

❸ Short designation of the parameter (from display B)

❹ Designation/function of the parameter

❺ Explanations on the parameter

❻ Number of parameter variant (from display A), scrolling with [F1]/[F3]. If no digit is indicated, a value must be entered for the variant. Initially, all variants and values are set to zero (exception: system parameter 1/**3** = 1)

❼ Short designation of the parameter variant (from display C); characters in italics indicate the input format of a value; parameters which cannot (yet) be activated are marked by "- - - - - - - -"

❽ Function of the variant

1. System parameters (1st level)

1	**1**	**Language**	Operating language
Determines the language for the display of texts			
0		german	texts are displayed in German
1		english	texts are displayed in English

1	**2**	**Pow.fail**	Power failure security
Specifies if the actual values and operating states are to be stored so that they will be available again after a power failure and when the equipment is 'normally' switched on			
0		n.secur.	no security
1		n.s.cal.	no security, but after powering-on, the equipment must first be calibrated (setting the reference measure, only with incremental encoders, refer also to 3/**91**)
2		security	security is provided
3		sec.cal.	security is provided, but powering-on, the equipment must first be calibrated (setting the reference measure, only with incremental encoders, refer also to 3/**91**)

1	**3**	**Unit1**	Configuration unit 1
Assignment of 1…6 axes to the 1st unit; only axes with subsequent numbers can be grouped starting with the axis always following the last one of the previous unit			
		X	input of the number of axes with the following possibilities: X = 1: 1 axis (1, default by manufacturer) X = 2: 2 axes (1 and 2, from here path control is possible, X = 3: 3 axes (1 to 3) cf. 2/**5**) X = 4: 4 axes (1 to 4, GEL 8610 only) X = 5: 5 axes (1 to 5, GEL 8610 only) X = 6: 6 axes (1 to 6, GEL 8610 only)

1	**4**	**Unit2**	Configuration unit 2
Assignment of 1…5 axes to the 2nd unit			
		X	input of the number of axes with the following possibilities: X = 0: no axis (assignment to the following units impossible) X = 1: 1 axis (2, 3, 4, 5, or 6) X = 2: 2 axes (2 and 3: path control possible) X = 3: 3 axes (2 to 4, 3 to 5 or 4 to 6, GEL 8610 only) X = 4: 4 axes (2 to 5 or 3 to 6, GEL 8610 only) X = 5: 5 axes (2 to 6, GEL 8610 only)

1	**5**	**Unit3**	Configuration unit 3
Assignment of 1…4 axes to the 3rd unit			
		X	input of the number of axes with the following possibilities: X = 0: no axis (assignment to the following units impossible) X = 1: 1 axis (3, 4, 5, or 6) X = 2: 2 axes (2 & 3 or 4 & 5 or 5 & 6, GEL 8610 only) X = 3: 3 axes (3 to 5 or 4 to 6, GEL 8610 only) X = 4: 4 axes (3 to 6, GEL 8610 only)

*continued for the **GEL 8310** further below*

1	**6**	**Unit4**	Configuration unit 4	GEL 8**6**10
Assignment of 1…3 axes to the 4th unit (GEL 8610 only)				
		X	input of the number of axes with the following possibilities: X = 0: no axis (assignment to the following units impossible) X = 1: 1 axis (4, 5, or 6) X = 2: 2 axes (4 & 5 or 5 & 6) X = 3: 3 axes (4 to 6)	

1	**7**	**Unit5**	Configuration unit 5	GEL 8**6**10
Assignment of 1 or 2 axes to the 5th unit (GEL 8610 only)				
		X	input of the number of axes with the following possibilities: X = 0: no axis (assignment to the following units impossible) X = 1: 1 axis (5 or 6) X = 2: 2 axes (5 & 6)	

1	**8**	**Unit6**	Configuration unit 6	GEL 8**6**10
Assignment of 1 axis to the 6th unit (GEL 8610 only)				
		X	input of the number of axes with the following possibilities: X = 0: no axis X = 1: 1 axis (6)	

1	**6** **9**	**Mem.card**	Memory card	GEL 8**3**10 GEL 8**6**10
Activation of a write protection for the optional memory card				
	0	n.protct	no protection, i.e., data can be stored any time (via F2 + resp.)	
	1	protect	data cannot be stored	

1	**7** **10**	**Card.fkt**	Special function with memory card	GEL 8**3**10 GEL 8**6**10
Determines if the equipment-specific axis parameters (AP) are not to be read from or written to the optional memory card; if the AP have never been stored on a certain memory card and you try to read the card <u>incl. AP</u>, then the message `mem.card error` is shown				
	0	excl. AP	axis parameters are not stored/overwritten	
	1	incl. AP	axis parameters are stored/overwritten	

1	**8** **11**	**Password**	Password inquiry for nominal value programming	GEL 8**3**10 GEL 8**6**10
Specifies if the programming mode for nominal values is to be accessible via a password only				
	0	inactive	no password inquiry	
	1	active	with password inquiry (define the password at the following storage location)	

1	9 / 12	**Password:**	Specification of the password	GEL 8**3**10 / GEL 8**6**10
Definition of the sequence of digits for the password of the programming mode for nominal values				
		XXXXXXXX	input of 1 to 8 digits	

1	10 / 13	**Term.P21**	Keyboard lock (terminal P21)	GEL 8**3**10 / GEL 8**6**10
Defines the function of the control input P21 (*/keyboard lock*) of the P1 module				
0		inactive	P21 without function	
1		keyboard	with Low level at P21 all keys are locked	
2		m.param	with Low level at P21 the programming mode for machine parameters is locked only (F2 + 1 without function)	

1	11 / 14	**Serial**	Serial interface	GEL 8**3**10 / GEL 8**6**10
Specifies the transmission rate for all serial interface types with LB2 protocol				
0		inactive	no serial data transmission	
1		1200 Bd	1,200 bits/sec	
2		2400 Bd	2,400 bits/sec	
3		4800 Bd	4,800 bits/sec	
4		9600 Bd	9,600 bits/sec	
5		19200 Bd	19,200 bits/sec	
6		28800 Bd	28,800 bits/sec (with RS 422/485 only)	
7		38400 Bd	38,400 bits/sec (with RS 422/485 only)	
8		48000 Bd	48,000 bits/sec (with RS 422/485 only; this rate is not supported by the most PCs)	
9		57600 Bd	57,600 bits/sec (with RS 422/485 only)	

<table>
<tr><td rowspan="2">1</td><td>12</td><td rowspan="2">Device #</td><td rowspan="2">Device number</td><td>GEL 8310</td></tr>
<tr><td>15</td><td>GEL 8610</td></tr>
<tr><td colspan="5">Specifies the device number (address) of the Controller for use with the RS 422/485</td></tr>
<tr><td colspan="2"></td><td>XX</td><td colspan="2">input of 1 or 2 digits (0 ... 31);
0 specifies a single Controller used</td></tr>
</table>

<table>
<tr><td rowspan="2">1</td><td>13</td><td rowspan="2">Protocol</td><td rowspan="2">Transmission protocol</td><td>GEL 8310</td></tr>
<tr><td>16</td><td>GEL 8610</td></tr>
<tr><td colspan="5">Specifies the transmission protocol to be used</td></tr>
<tr><td colspan="2">0</td><td>LB2</td><td colspan="2">standard protocol</td></tr>
<tr><td colspan="2">1</td><td>Terminal</td><td colspan="2">optional protocol for operating terminal GEL 131;
this variant can only be selected if the appropriate option has been ordered</td></tr>
</table>

<table>
<tr><td rowspan="2">1</td><td>14</td><td rowspan="2">Service</td><td rowspan="2">Deactivation of passwords</td><td>GEL 8310</td></tr>
<tr><td>17</td><td>GEL 8610</td></tr>
<tr><td colspan="5">All passwords of the programming mode for machine parameters and for the display of signal states can be deactivated while servicing</td></tr>
<tr><td colspan="2">0</td><td>inactive</td><td colspan="2">standard mode, with password inquiry</td></tr>
<tr><td colspan="2">1</td><td>active</td><td colspan="2">service mode, no password inquiry</td></tr>
</table>

1	**15** **18**	**Reserve**		GEL 8**3**10 GEL 8**6**10
		• • •	*not used for the standard version*	
1	**21** **24**	**Reserve**		GEL 8**3**10 GEL 8**6**10
		- - - - - - - -		

Change to the next programming level via ESC ESC ⬇/⬆ ➡

2. **Unit parameters** (2nd level)

2	**1**	**Batch/t**	number of pieces (batch counter) and auto start
Specifies whether the ‘number of pieces’ nominal value type is to be part of a sentence and/or whether individual times for generating an automatic start signal are to be preset. If one of the variants 2 or 3 has been activated the time preset by the unit parameter 2/**10** will be ignored.			
0		inactive	no number of pieces and no auto start time/sentence
1		batch	with number of pieces
2		t auto	time specification for auto start with each sentence The input format of the times in the nominal value programming mode is the same as that of storage location 2/**10**; with **0** this sentence will not be automatically started.
3		both	both possibilities

2	**2**	**M. func.**	Machine functions
Specifies whether the ‘machine functions’ nominal value type is to be part of a sentence or not, and, if so, at which optional data output module the **m**achine **f**unctions (=**MF**) are to be output (see section 4.10.2 and storage location 2/**11**). With 8 MF, input is **binary** (10010011 = MF8/5/2/1) and with 24 MF it is **octal** (77777777 = all MF; 273 = MF8/6/5/4/2/1).			
0		no	no machine functions
1		8 out1.0	8 MF parallel at the 1st data output, decades 10^0+10^1
2		8 out1.2	8 MF parallel at the 1st data output, decades 10^2+10^3
3		8 out1.4	8 MF parallel at the 1st data output, decades 10^4+10^5
4		8 out2.0	8 MF parallel at the 2nd data output, decades 10^0+10^1
5		8 out2.2	8 MF parallel at the 2nd data output, decades 10^2+10^3
6		8 out2.4	8 MF parallel at the 2nd data output, decades 10^4+10^5
7		8 out3.0	8 MF parallel at the 3rd data output, decades 10^0+10^1
8		8 out3.2	8 MF parallel at the 3rd data output, decades 10^2+10^3
9		8 out3.4	8 MF parallel at the 3rd data output, decades 10^4+10^5

continued on the next page ➔

GEL 8310 only: *continuation 2/**2***

10	24 out 1	24 MF parallel at the 1st data output (R module: 12 MF)
11	24 out 2	24 MF parallel at the 2nd data output (R module: 12 MF)
12	24 out 3	24 MF parallel at the 3rd data output (R module: 12 MF)

GEL 8610 only:

10	8 out4.0	8 MF parallel at the 4th data output, decades 10^0+10^1
11	8 out4.2	8 MF parallel at the 4th data output, decades 10^2+10^4
12	8 out4.4	8 MF parallel at the 4th data output, decades 10^4+10^5
13	24 out 1	24 MF parallel at the 1st data output (R module: 12 MF)
14	24 out 2	24 MF parallel at the 2nd data output (R module: 12 MF)
15	24 out 3	24 MF parallel at the 3rd data output (R module: 12 MF)
16	24 out 4	24 MF parallel at the 4th data output (R module: 12 MF)

2	**3**	**Speed**	Speed rate
Specifies whether the ‘speed rate’ nominal value type (for each assigned axis) is to be part of a sentence or not (refer also to section 4.2)			
0		no	no speed rate preset in the sentence (value from 3/**34**)
1		yes (➧1)	speed rate is to be set for each sentence, and mode no. 1 with continuous sentence processing (refer to section 4.3)
2		yes (➧2)	speed rate is to be set for each sentence, and mode no. 2 with continuous sentence processing (refer to section 4.3)
3		yes (➧3)	only in connection with linear path control (storage location 2/**5** = »linear«; otherwise, a parameter error is generated): speed rate specification in the sentence with continuous sentence processing and spline (refer to section 4.13); internal start as for mode 1

2	**4**	**Text**	Identification text
Specifies whether identification text is to be part of a sentence			
0		- - - - - - - -	*function presently not available yet*

2	**5**	**Interp.**	Interpolation (path control)
Specifies if a path control is to be used for the assigned axes when positioning (cf. section 4.13), or if an optional function is to be used (e.g. synchro control option). A new programming will cause the deletion of all nominal value programs of this unit (with safety inquiry).			
0	inactive		no path control
1	linear		linear path control for the following axes: • 1 … 3 / 6 for unit 1 • 2 and 3 for unit 2

2	**6**	**Program**	Program selection
Specifies if the program is to be selected via the keyboard, via one of the optional data inputs (BCD or binary, refer to section 4.10.1), or via the serial interface. In all cases, 0 must be programmed at storage location 2/**7** as this is evaluated with priority. A program selection can principally be made only in the **reset** state of the Automatic mode; specified data are taken over with the next start signal.			
0	keyboard		program selection via terminal, key combination F3 + 1
1	i1.0 bcd		BCD program selection via the 1st data input E1, decades 10^0 and 10^1
2	i1.2 bcd		BCD program selection via the 1st data input E1, decades 10^2 and 10^3
3	i1.4 bcd		BCD program selection via the 1st data input E1, decades 10^4 and 10^5
4	i2.0 bcd		BCD program selection via the 2nd data input E2, decades 10^0 and 10^1
5	i2.2 bcd		BCD program selection via the 2nd data input E2, decades 10^2 and 10^3
6	i2.4 bcd		BCD program selection via the 2nd data input E2, decades 10^4 and 10^5
7	i3.0 bcd		BCD program selection via the 3rd data input E3, decades 10^0 and 10^1
8	i3.2 bcd		BCD program selection via the 3rd data input E3, decades 10^2 and 10^3
9	i3.4 bcd		BCD program selection via the 3rd data input E3, decades 10^4 and 10^5

continued on the next page ➔

GEL 8<u>3</u>10 only: *continuation 2/**6***

10	serial	program selection via the serial interface

GEL 8<u>6</u>10 only:

10	i4.0 bcd	BCD program selection via the 4th data input E4, decades 10^0 and 10^1
11	i4.2 bcd	BCD program selection via the 4th data input E4, decades 10^2 and 10^3
12	i4.4 bcd	BCD program selection via the 4th data input E4, decades 10^4 and 10^5
13	serial	program selection via the serial interface

2	**7**	**Sentence**	Sentence/program selection
Specifies if a sentence of a certain program is to be activated via one of the optional data inputs (refer to section 4.10.1); new data for the sentence and program are taken over with the next start signal after the current sentence has been processed (number of pieces!)			
0		program	no sentence selection, the processing of the sentence is determined by the normal program execution (program selection according to storage location 2/**6**)
1		input 1	sentence/program selection via the 1st data input E1
2		input 2	sentence/program selection via the 2nd data input E2
3		input 3	sentence/program selection via the 3rd data input E3

GEL 8<u>6</u>10 only:

4	input 4	sentence/program selection via the 4th data input E4

2	**8**	**Sent.out**	Sentence/program number to data output
Specifies at which of the optional data outputs the numbers of the current sentence and program are to be output			
0		inactive	no output of the sentence/program number
1		output 1	sentence/program number to the 1st data output
2		output 2	sentence/program number to the 2nd data output
3		output 3	sentence/program number to the 3rd data output

GEL 8610 only:

4	output 4	sentence/program number to the 4th data output

2	**9**	**Sig. out**	Program processing signals to data output
Specifies whether the program processing signals (refer to section 4.5.5) are to be output and if so at which optional data output; refer also to storage location 2/**11**			
0		inactive	no signal output
1		out 1.0	1st data output, decade 10^0
2		out 1.1	1st data output, decade 10^1
3		out 1.2	1st data output, decade 10^2
4		out 1.3	1st data output, decade 10^3
5		out 1.4	1st data output, decade 10^4
6		out 1.5	1st data output, decade 10^5
7		out 2.0	2nd data output, decade 10^0
8		out 2.1	2nd data output, decade 10^1
9		out 2.2	2nd data output, decade 10^2
10		out 2.3	2nd data output, decade 10^3
11		out 2.4	2nd data output, decade 10^4
12		out 2.5	2nd data output, decade 10^5
13		out 3.0	3rd data output, decade 10^0
14		out 3.1	3rd data output, decade 10^1
15		out 3.2	3rd data output, decade 10^2
16		out 3.3	3rd data output, decade 10^3
17		out 3.4	3rd data output, decade 10^4
18		out 3.5	3rd data output, decade 10^5

continued on the next page ➔

GEL 8<u>6</u>10 only: *continuation 2/**9***

19	out 4.0	4th data output, decade 10^0
20	out 4.1	4th data output, decade 10^1
21	out 4.2	4th data output, decade 10^2
22	out 4.3	4th data output, decade 10^3
23	out 4.4	4th data output, decade 10^4
24	out 4.5	4th data output, decade 10^5

2	**10**	**t auto**	Auto start
Once the signal *actual=nominal is* given for all axes of the unit, the time t_{auto}. is active. After the programmed value has been counted down, a start signal is internally generated for the unit. If, during this time, an axis exits the nominal position tolerance range, the timer is reset and thus no start signal is generated (refer also to storage location 2/**1**).			
		XX.XX	value range : 0...99.99 s at **0** the function is inactive

2	**11**	**Out.func**	Mode for signal output
Specifies if the machine functions and/or program processing signals (*sentence end*, *block end*, and *program end*; refer to section 4.5.5) are to be output with the start signal of the sentence (standard) or only if the *actual=nominal* signal is active for <u>all</u> axes within the unit			
0		inactive	standard output (with s*tart*)
1		m. func.	machine functions with *actual=nominal*
2		sig. out	program processing signals with *actual=nominal*
3		Mf/Sign.	machine functions and program processing signals with *actual=nominal*

2	**12**	**ResetOut**	Program processing signal ‘reset’
Specifies whether the *reset* signal is to be output and if so which other program processing signal is to be dropped instead (refer to section 4.5.5)			
0		inactive	*reset* is not used
1		sent.end	*reset* is used instead of *sentence end*
2		blockEnd	*reset* is used instead of *block end*

2	**13** **14**	**Reserve**	
not used for the standard version			
		- - - - - - - -	

Remarks:

3. **Axis parameters** (3rd level)

3	**1**	**Encoder**	Actual value adjustment
Specification of the used encoder and setting of the edge evaluation of the 0° and 90° tracks at the count input and the coding of the connected encoder (refer also to section 5.2)			
0		incr. x1	incremental encoder with 1-fold edge evaluation (nominal pulse number)
1		incr. x2	incremental encoder with 2-fold edge evaluation (double nominal pulse number)
2		incr. x4	incremental encoder with 4-fold edge evaluation (quadruple nominal pulse number)
3		Gray	absolute encoder, Gray code
4		/Gray	parallel absolute encoder, Gray code with logic reversal
5		binary	parallel absolute encoder, binary code
6		/binary	parallel absolute encoder, binary code with logic reversal
7		BCD	parallel absolute encoder, BCD
8		/BCD	parallel absolute encoder, BCD with logic reversal
9		SSI 25b	25 bits multi-turn absolute encoder with serial data output Gray code, refer to storage location 3/**54**
10		SSI 13b	13 bits single-turn absolute encoder with serial data output, Gray code, refer to storage location 3/**54**

3	**2**	**Directio**	Count direction
Inverting of actual value counting direction WARNING **It is mandatory to read sections 5.4 and 4.1.3 before performing any changes!**			
0		no rever	no reversal of the count direction
1		reversal	reversal of the count direction

3	3	**Multipl.**	Multiplier
Multiplier for the encoder input (for incremental encoders after edge evaluation); refer to section 5.2			
		XX.XXXX	Input of a value ≥ 0 and ≤ 99.9999 **0** ≡ 1.0000

3	4	**Disp.mul**	Multiplier for actual value display
Multiplier for the display of all values in actual measuring units (displays A and C); the setting is only effective if at least one decimal place has been programmed (3/**5** > 0)			
0		x1	display unchanged
1		x0.1	the display is shifted to the right by one digit, i.e., the last digit is removed from the display (example: with »x1« 123.45, with »x0.1« 123.4)

3	5	**DecPoint**	Decimal point
Decimal places (resolution) of the nominal and actual position/length values in the display and at the input, in the programming mode of machine parameters, this also applies to the speed rates (cf. storage locations 3/**45** and 3/**46**); refer to section 5.2			
0		X.	no decimal place
1		X.X	one decimal place
2		X.XX	two decimal places
3		X.XXX	three decimal places
4		X.XXXX	four decimal places

3	6	**Corr.val**	Correction value
Correction value for the compensation of cutting losses (with system of incremental dimensions) or tool wear (with system of absolute dimensions); the value programmed may be changed via direct entry in the Automatic mode (refer to storage location 3/**7** and section 4.7			
0		*XXXXXXXX*	max. 8 digits incl. sign (-) and decimal point; value in actual measuring units

3	**7**	**Man.corr**	Direct entry of a correction value
A correction value can be entered after pressing [F1]+[3] / [↤] in the reset state of the Automatic mode; it overwrites the originally programmed value in storage location 3/**6**; refer to section 4.7			
0	inactive	direct entry is disabled	
1	active	direct entry is enabled	

3	**8**	**Ro.table**	Rotary table
Restriction of the counting range for rotary table applications			
	XXXXXXXX	max. 8 digits incl. decimal point, positive values only; for the input some special characteristics have to be considered, refer to section 4.8	

3	**9**	**Man.cal.**	Manual calibration functions
Certain calibration functions may be performed via the keyboard in the Automatic mode; refer to section 4.6			
0	inactive	no calibration function via the keyboard	
1	ref.val.	direct entry of a reference measure via [F1]+[1] / [⌖]; the nominal value overwrites the originally programmed value in storage location 3/**15** or 3/**85**	
2	auto cal	set the *search for reference* signal via the keyboard ([F1]+[2] / [◎])	
3	val/auto	both variants 1 and 2 are enabled	

3	**10**	**Set ref.**	Setting of the reference measure
Specifies the direction of travel for the setting of the reference measure **when positioning the drive;** refer to section 4.6			
0		inactive	the setting of the reference measure is disabled
1		forward	the reference measure will be set when moving the drive in the forward direction
2		reverse	the reference measure will be set when moving the drive in the reverse direction
3		forw/rev	the reference measure will be set irrespective of the direction of travel

3	**11**	**Auto cal**	Automatic reference search routine
Specifies the direction of motion for setting the reference measure for the **automatic reference search routine** (auto calibration). This can only be initiated in the reset state of the unit and if High level is applied to the corresponding stop input; refer to section 4.6			
0		inactive	search routine is disabled
1		autoforw	setting the reference measure in the forward direction
2		auto rev	setting the reference measure in the reverse direction

3	**12**	**Fine**	*Reference fine* signal
Specifies the switching direction of the *reference fine* signal; refer to section 4.6			
0		↓ edge	negative signal edge (High → Low) is evaluated
1		↑ edge	positive signal edge (Low → High) is evaluated
2		↑↓ edge	both signal edges are evaluated

3	**13**	**Coarse**	*Reference coarse* signal
Specifies the switching state of the *reference coarse* signal; refer to section 4.6			
0		low	Low level ≙ logical 1
1		high	High level ≙ logical 1

3	**14**	**R.switch**	*Reversing switch* signal
Specifies the switching level of the *reversing switch* signal for the automatic reference search routine; refer to section 4.6			
0		low	Low level ≙ logical 1
1		high	High level ≙ logical 1

3	**15**	**Ref.val.**	Value of the 1st reference measure
Under the precondition that the *reference2/1* signal is <u>not</u> active (Low level at P22), the actual value is set to this value as soon as the drive exceeds the reference point; the here specified value can also be changed via direct entry in the Automatic mode under the above stated conditions (refer to storage location 3/**9** and section 4.6, 2nd reference measure: 3/**85**)			
		XXXXXXXX	max. 8 digits incl. sign (-) and decimal point; specification of the value in actual measuring units

3	**16**	**Cal. spd.**	Auto calibration speed rate
Speed rate of the automatic reference search routine up to the reversal point (proximity switch); refer to section 4.6			
		XXXXXX	max. 6 digits incl. decimal point; value in actual measuring units per second

3	**17**	**Rev.spd.**	Reversing speed rate
Speed rate of the automatic reference search routine from the reversal point (proximity switch) towards the reference point; refer to section 4.6			
		XXXXXX	max. 6 digits incl. decimal point; value in actual measuring units per second

3	**18**	**ManOper.**	Operating state for manual positioning
Specifies the operating state in which the drive can manually be moved; High level must be applied to the corresponding */stop* input (P4/7/10)			
0		reset	the Controller must be in the reset state
1		stop/res	the Controller must be in the reset or interrupted state

3	**19**	**Man.ctrl**	Manual positioning
Specifies the data input (refer to section 4.10.1) and the position to which the control signals for the manual positioning of the drive are applied (refer to appendix B), or if the controlling shall be performed via the keyboard			
0		inactive	manual controlling is disabled
1		input1.0	positioning via the 1st data input E_1, decade 10^0
2		input1.1	positioning via the 1st data input E_1, decade 10^1
3		input1.2	positioning via the 1st data input E_1, decade 10^2
4		input1.3	positioning via the 1st data input E_1, decade 10^3
5		input1.4	positioning via the 1st data input E_1, decade 10^4
6		input1.5	positioning via the 1st data input E_1, decade 10^5
7		input2.0	positioning via the 2nd data input E_2, decade 10^0
8		input2.1	positioning via the 2nd data input E_2, decade 10^1
9		input2.2	positioning via the 2nd data input E_2, decade 10^2
10		input2.3	positioning via the 2nd data input E_2, decade 10^3
11		input2.4	positioning via the 2nd data input E_2, decade 10^4
12		input2.5	positioning via the 2nd data input E_2, decade 10^5
13		input3.0	positioning via the 3rd data input E_3, decade 10^0
14		input3.1	positioning via the 3rd data input E_3, decade 10^1
15		input3.2	positioning via the 3rd data input E_3, decade 10^2
16		input3.3	positioning via the 3rd data input E_3, decade 10^3
17		input3.4	positioning via the 3rd data input E_3, decade 10^4
18		input3.5	positioning via the 3rd data input E_3, decade 10^5

continued on the next page ➔

GEL 8<u>3</u>10 only: *continuation 3/**19***

19	keyboard	positioning via the keyboard ([F1]/[F2]+[⬆]/[⬇]) in the Automatic mode and during teach-in operation

GEL 8<u>6</u>10 only:

19	input4.0	positioning via the 4th data input E4, decade 10^0
20	input4.1	positioning via the 4th data input E4, decade 10^1
21	input4.2	positioning via the 4th data input E4, decade 10^2
22	input4.3	positioning via the 4th data input E4, decade 10^3
23	input4.4	positioning via the 4th data input E4, decade 10^4
24	input4.5	positioning via the 4th data input E4, decade 10^5
25	keyboard	positioning via the keyboard ([✋]+[<]/[>]/[<<]/[>>]) in the Automatic mode and during teach-in operation

3	**20**	**Man.pol.**	Polarity for manual drive control
Assignment of the positioning direction to the appropriate signals at the optional data input connector or to the keys for the manual positioning			
0		▶ = forw	positive voltage *) and *forward* signal for signals and keys [⬇], [>], [>>]
1		◀ = forw	positive voltage *) and *forward* signal for signals and keys [⬆], [<], [<<]

*) The voltage can also be negative depending on the programming of storage location 3/**25**.

3	**21**	**▶ speed**	Slow speed, forward
	22	**▶▶ speed**	Fast speed, forward
	23	**◀ speed**	Slow speed, reverse
	24	**◀◀ speed**	Fast speed, reverse
Speed rates for the manual positioning			
		XXXXXX	max. 6 digits incl. decimal point; value in actual measuring units per second (e.g. ‘100’ for 100.00 mm/sec)

3	**25**	**Polar. O**	Polarity of the analog output
Assignment of the voltage polarity to the direction of motion			
WARNING	The **polarity** may only **be changed** if the drive does not move mechanically in the desired direction although the electric connection of the drive assembly (amplifier, motor, tacho generator) is **correct** (it is mandatory to read sections 4.1.3 and 5.4.)		
0	+ = forw	positive voltage for forward motion	
1	– = forw	negative voltage for forward motion	

3	**26**	**Analog O**	Voltage range of the analog output
Specifies whether the analog output voltage is to be bipolar or unipolar			
0	+/–10 V	output voltage is bipolar (the sign depends on the direction of travel)	
1	+10 V	output voltage is unipolar (direction of travel only via the appropriate *forward / reverse* signals at terminal strip D); here, the programming of storage location 3/**25** is without any effect	

3	**27** **28**	**S dead +** **S dead –**	Positive dead range Negative dead range
As long as the drive is within the range fixed by S_{dead+} and S_{dead-} around the nominal position (after the internal control pre-set has reached the nominal position value), the analog output remains switched off, i.e., 0.000 V is output. This also applies to any (idle) position for activated closed loop position control (refer to storage location 3/**47** and section 4.4.1). S_{dead+}: 'nominal – actual' difference > 0, S_{dead-}: 'nominal – actual' difference < 0 Depending on the multiplier **m** (storage location 3/**3**), the following condition is valid: $S_{dead} \geq 1$ for $m \leq 1$, $S_{dead} \geq 2$ for $1 < m \leq 2$, $S_{dead} \geq 3$ for $2 < m \leq 3$ etc.			
	XXXXXXXX	max. 8 digits incl. decimal point, only positive; value in actual measuring units	

3	**29** **30**	**Umin +** **Umin –**	Minimum positive voltage Minimum negative voltage
Minimum positive/negative voltage for the drive amplifier at which it can still control the drive (in both forward and reverse direction); refer to section 5.5			
		XX.XXX	range of values: 0...(+)10,000 V, resolution is 1 mV

3	**31**	**Umax**	Maximum voltage
Maximum voltage for the drive amplifier generating the admissible maximum speed of the drive (for both directions)			
		XX.XXX	range of values: 0...(+)10,000 V, resolution is 1 mV

3	**32**	**MaxSpeed**	Maximum speed
Max. absolute speed rate (v_{max}) of the drive at U_{max}; refer to section 5.6.1			
0		*XXXXXX*	max. 6 digits incl. decimal point; value in actual measuring units per second, only positive

3	**33**	**Ksp**	Control factor
This K_{sp} factor specifies the dynamic range of the drive control; refer to section 5.6.2			
		XXX.X	range of values: 0...999.9, the dimension is 1/sec at **0** the closed loop position control is switched off, i.e., the drive is positioned only by the speed pre-control

3	**34**	**Speed**	Working speed rate
Specification of the working speed rate for the positioning processes; this value is used if the nominal value type ‘speed’ is not part of a sentence for the associated unit, i.e., no nominal values are preset for the speed rate per sentence. WARNING: For internal calculation reasons, a value must be programmed here **also** if presetting the speed rate in the sentence. This value must be <u>higher than or equal to the used maximum nominal value</u> since otherwise the times for accelerating and braking could unexpectedly increase thus **causing a potential danger**.			
		XXXXXX	max. 6 digits incl. decimal point; value in actual measuring units per second, only positive

3	**35** **36** **37** **38**	**t accel+** **t accel-** **t brake+** **t brake-**	Max. acceleration time, forward Max. acceleration time, reverse Max. braking time, forward Max. braking time, reverse
The times t_{accel+} ... t_{brake-} result from the maximum value of acceleration or braking a_+ ... b_- in the respective direction (+ ≙ forward = positive count direction) and are to be deducted from the technical data of the drive system: $t_{accel+/-} = v_{max}/a_{+/-}$, $t_{brake+/-} = v_{max}/b_{+/-}$ (refer to sections 5.3 and 5.6.3). Programming of the storage locations 3/**36** ... 3/**38** is not mandatory. The value stored at 3/**35** (t_{accel+}) also applies to acceleration in reverse direction and braking in both directions; if the storage location 3/**38** = 0 then 3/**36** is effective for t_{brake-} (or 3/**35** if 3/**36** = 0).			
		X.XXX	range of values: 0...9.999 sec For internal calculation reasons, the following restriction applies: $t_{accel,\ brake} \leq (v_{max} * axes^2) / 0.03$ $t_{accel,\ brake}$ in milliseconds v_{max} in DispU/sec (value from 3/**32** without decimal point) axes = number of activated axes (1 up to 3 / 6) If the value entered is too high, it is internally reduced to the above maximum value.

3	**39**	**t jerk**	Jerking time
The jerk determines the positioning characteristic of the drive; it is defined by the time t_{jerk} in which the maximum acceleration is reached (jerk = a_+/t_{jerk} = $v_{max}/(t_{accel+} * t_{jerk})$; the larger t_{jerk} the smoother the accelerating and braking process; refer to sections 4.4.2 and 5.6.4			
		X.XXX	range of values: 0...9.999 sec For internal calculation reasons, the following restriction applies $t_{jerk} \leq (v_{max} * axes^3) / (0.03 * t_{accel+})$ t_{jerk} in milliseconds v_{max} in DispU/sec (value from 3/**32** without decimal point) axes = number of activated axes (1 up to 3 / 6) t_{accel+} in milliseconds (value from 3/**35** without decimal point) If the value entered is too high, it is internally reduced to the above maximum value.

3	**40** **41**	**Tol. +** **Tol. –**	Positive tolerance Negative tolerance
To keep the *actual=nominal* signal stable, a tolerance range may be specified in which the signal is output; Tol.+: 'nominal – actual' difference is positive, Tol.–: 'nominal – actual' difference is negative			
		XXXXXXXX	max. digits incl. decimal point, only positive; value in actual measuring units

3	**42** **43**	**S max +** **S max –**	Max. positive contouring error Max. negative contouring error
Once the 'nominal – actual' difference exceeds the specified values, – */fault* signal is output (level at terminal K1 changes from High to Low), – the Controller changes into the stop state (sentence number flashes[*)]) or into the reset state (with manual positioning or reference search routine). This state is acknowledged with a *start* or *search for reference* signal. [*)] Can only be recognized if the triggering axis is displayed.			
	0	*XXXXXXXX*	max. digits incl. decimal point, only positive; value in actual measuring units; with **0** the contouring error monitoring can be deactivated but this is not recommended for **safety reasons!**

3	**44**	**Measure**	Measurement system
Specifies the measurement system, the Controller shall work with			
0		absolute	system of absolute dimensions (fixed zero processing): position values entered are absolute positions; if in the programming mode for nominal values you switch over to lengths (F3 + 0 / ⊕) incremental dimension processing can be obtained for individual sentences
1		relative	system of incremental dimensions (floating zero processing): position values entered are relative lengths (with each start the actual value is set to 0); it is not possible to switch over to the absolute dimension processing for individual sentences within the programming mode for nominal values (F3 + 0 / ⊕)
2		residual	as above, including, however, the computation of residual values for the compensation of positioning inaccuracies (with next start the actual value counter is set with the *actual – nominal* difference value of the previous sentence)

3	**45**	**Spd.mult**	Multiplier for speed rate values
The nominal values of the speed rate in the sentence can be entered with another measuring unit than the one actually used for the equipment (actual measuring units per second); the appropriate adaptation is effected by »Spd.mult«, refer to section 4.2			
		XX.XXXX	input of a value ≥ 0 and ≤ 99.9999 **0** ≡ 1.0000 (standard)

3	**46**	**DecP.spd**	Decimal point for speed rate values
Decimal places of the nominal speed rate values according to the specification for the calculation of the »Spd.mult« multiplier (refer to section 4.2); if the standard measuring unit is to be used »DecP.spd« must be programmed as under storage location 3/**5**			
0		X.	no decimal places
1		X.X	one decimal place
2		X.XX	two decimal places
3		X.XXX	three decimal places
4		X.XXXX	four decimal places

3	**47**	**Reg.stop**	Control in the stop/reset state
Specifies if the closed loop position control is to be active in the interrupted (stop) or reset state			
0		inactive	closed loop control is switched off, the *release brake* and *cancel controller lock* signals are reset
1		active	closed loop control is active, the signals *release brake* and *cancel controller lock* remain active; in this case storage locations 3/**48** to 3/**50** are internally set to 1, too, irrespective of the actual programming

3	**48**	**RegStart**	Control conditions after reaching the nominal position
In the started state, the closed loop position control always remains active; it has, however, to be determined if the *release brake* and *cancel controller lock* signals are to be reset once the nominal position is reached			
0		with sig.	signals are reset (only valid, if storage location 3/**47** = 0)
1		w/o sig.	signals remain active (automatically set with 3/**47** = 1)

3	**49**	**Reg.cal.**	Control of the automatic calibration
Specifies if the reference search routine is to be closed-loop-controlled or just speed-controlled (calculated voltage curve)			
0		inactive	without closed loop control (only valid, if storage location 3/**47** = 0)
1		active	with closed loop control (automatically set with 3/**47** = 1)

3	**50**	**Reg.man.**	Control of the manual positioning
Specifies whether the manual positioning of the drive is to be closed-loop-controlled or just speed-controlled (calculated voltage curve)			
0		inactive	without closed loop control (only valid, if storage location 3/**47** = 0)
1		active	with closed loop control (automatically set with 3/**47** = 1)

3	**51** **52**	**tb open** **tb close**	Brake opening time Brake closing time
Brake setting time measured from the *release brake* signal up to the final release (3/**51**) or from resetting the signal up to the final engaging of the brake (3/**52**); after $t_{b\ open}$ has expired, the positioning control is activated (start delay); after $t_{b\ close}$ has expired, the signal *cancel controller lock* is reset and the closed loop position control becomes inactive (control stop delay); refer to section 4.5.4. The higher time value of those specified for each axis is used if several axes are path controlled.			
		X.XX	range of values: 0...9.99 sec

3	**53**	**AE-zero**	Zero adjustment for angle encoders (AE)
Zero shift of the actual value The value to be read for the mechanical zero position (origin) of the drive has to be entered with reverse sign. After the mechanical zero position has been changed, the programmed value must be erased first before a readjustment is made; refer to section 5.2.2.			
		XXXXXXXX	max. digits incl. sign (-) and decimal point; value in actual measuring units

3	**54**	**AE-bits**	Resolution of the absolute encoder (AE)
Number of bits (data lines) used of the installed AE; for AEs with serial data output (SSI): resolution of the **single-turn part**			
		XX	max. 2 input digits

3	**55**	**AE-e'ble**	Enable signal for the absolute encoder (AE)
Specifies the logic level of the *enable* signal (the *latch* signal is always active-High for all connected AE) **Important:** The used AE must make its data available **within 100 µsec** after the *enable* signal has become active.			
0		low	Low level ≙ logic 1
1		high	High level ≙ logic 1

3	**56**	**Park.fct**	Parking function
Specifies the mode for moving the to the park position; refer to section 4.9			
0		inactive	machine does not move to the park position; direct entry of a position value is disabled
1		➧ batch	machine moves to the park position by a *start* signal before the batch counter is incremented (1st, 3rd, 5th, ... start)
2		batch ➧	machine moves to the park position by a *start* signal after the batch counter has been incremented (2nd, 4th, 6th, ... start)
3		➧ sentence	machine moves to the park position by a *start* signal at the beginning of a new sentence
4		sentence ➧	machine moves to the park position by a *start* signal after the processing of the current sentence
5		➧ cycle	machine moves to the park position by a *start* signal at the beginning of a new program cycle
6		cycle ➧	machine moves to the park position by a *start* signal at the end of a program cycle (after reaching the nominal number of pieces preset in the last sentence of the program)

3	**57**	**Dir.park**	Direct entry of a park position
A value for the park position can be entered by actuating the [F1]+[4] keys in the Automatic mode; it will overwrite the programmed value at storage location 3/**58**			
0		inactive	direct entry disabled
1		active	direct entry enabled (precondition: 3/**56** is not »inactive«)

3	**58**	**Park.pos**	Value of the park position
The position value programmed at this location can also be overwritten via direct entry in the Automatic mode (refer to storage location 3/**57**)			
		XXXXXXXX	max. digits incl. sign (-) and decimal point; value in actual measuring units

3	**59**	**Park.spd**	Parking speed rate
Speed rate for moving to the park position			
		XXXXXX	max. 6 digits incl. decimal point, only positive; value in actual measuring units per second

3	**60**	**ParkMfct**	Machine functions for parking
The machine functions programmed here are available while moving to the park position and staying there (as long as the Controller is in the started state); the output is made as determined by the unit parameter 2/**2** (see there). The parking machine functions are principally unit-related. They have, however, one special feature: the programmed machine functions of all axes which are **assigned to one unit** are logically OR-ed. Example with 2 axes and 8 machine functions (binary): $ParkMfct_{axis1}$ = 1, $ParkMfct_{axis2}$ = 1100 ⇒ output: 00001101			
		XXXXXXXX	max. 8 digits, binary or octal (refer to storage location 2/**2**)

3	**61**	**Ranges**	Output of range signals
Specifies if range signals are to be output (refer to section 4.5.6) and, if so, at which optional data output (refer to section 4.10.2)			
0		inactive	no range signal output
1		out 1.0	1st data output, decade 10^0
2		out 1.1	1st data output, decade 10^1
3		out 1.2	1st data output, decade 10^2
4		out 1.3	1st data output, decade 10^3
5		out 1.4	1st data output, decade 10^4
6		out 1.5	1st data output, decade 10^5
7		out 2.0	2nd data output, decade 10^0
8		out 2.1	2nd data output, decade 10^1
9		out 2.2	2nd data output, decade 10^2
10		out 2.3	2nd data output, decade 10^3
11		out 2.4	2nd data output, decade 10^4
12		out 2.5	2nd data output, decade 10^5
13		out 3.0	3rd data output, decade 10^0
14		out 3.1	3rd data output, decade 10^1
15		out 3.2	3rd data output, decade 10^2
16		out 3.3	3rd data output, decade 10^3
17		out 3.4	3rd data output, decade 10^4
18		out 3.5	3rd data output, decade 10^5

GEL 8610 only:

19	out 4.0	4th data output, decade 10^0
20	out 4.1	4th data output, decade 10^1
21	out 4.2	4th data output, decade 10^2
22	out 4.3	4th data output, decade 10^3
23	out 4.4	4th data output, decade 10^4
24	out 4.5	4th data output, decade 10^5

3	**62**	**RangeFct**	Function for range signals
Specifies if the values programmed for ranges R1 to R4 (start, end) are to be absolute or relative positions or if the range signals are to be used for the control of fast/slow-speed drives (the individual functions are described in section 4.5.6)			
0		absolute	the values identify absolute (actual) positions, the signals can be output in any operating mode and in any operating state of the Controller
1		relative	the values identify relative positions in relation to the nominal position (‘nominal – actual’ distance), the signals are only output during a positioning process (in the started state)
2		driveSig	as above, signals are, however, negated and used for the control of fast/slow-speed drives; the ranges have the following meaning: R1: slow speed (linked to R2) R2: fast speed R3: forward direction R4: reverse direction For further explanations see section 4.5.6, under ‘Drive signals’.

3	**63** **64**	**R1: Beg.** **R1: End**	Start value of range R1 End value of range R1
Specifies the position of the 1st range R1 (absolute or relative to the nominal position, refer to storage location 3/**62**) signal output: *Beg.* < *End* ⇒ High level starting with *Beg.* to *End* –1 *Beg.* = *End* ⇒ no output (Low level) *Beg.* > *End* ⇒ inverted output: Low level starting with *End* to *Beg.*–1 For the drive control (3/**62** = 2): R1 = ***slow speed*** signal The signal level of R1 is inverted and internally linked to range signal R2 (refer to section 4.5.6).			
		XXXXXXXX	max. digits incl. sign (-) and decimal point; value in actual measuring units

3	**65** **66**	**R2: Beg.** **R2: End**	Start value of range R2 End value of range R2

Specifies the position of the 2nd range R2 (absolute or relative to the nominal position, refer to storage location 3/**62**)

signal output: refer to storage location 3/**63**/**64**

For the drive control (3/**62** = 2): R2 = ***fast speed*** signal

The signal level of R2 is inverted (refer to section 4.5.6).

	XXXXXXXX	max. digits incl. sign (-) and decimal point; value in actual measuring units

3	**67** **68**	**R3: Beg.** **R3: End**	Start value of range R3 End value of range R3

Specifies the position of the 3rd range R3 (absolute or relative to the nominal position, refer to storage location 3/**62**)

signal output: refer to storage location 3/**63**/**64**

For the drive control (3/**62** = 2): R3 = ***forward*** signal

The end value is internally set to the maximum positive value possible and can neither be indicated nor changed; a value programmed here will be ignored (refer to section 4.5.6).

	XXXXXXXX	max. digits incl. sign (-) and decimal point; value in actual measuring units

3	**69** **70**	**R4: Beg.** **R4: End**	Start value of range R4 End value of range R4

Specifies the position of the 4th range R4 (absolute or relative to the nominal position, refer to storage location 3/**62**)

signal output: refer to storage location 3/**63**/**64**

For the drive control (3/**62** = 2): R4 = ***reverse*** signal

The end value is internally set to the maximum negative value possible and can neither be indicated nor changed; a value programmed here will be ignored (refer to section 4.5.6).

	XXXXXXXX	max. digits incl. sign (-) and decimal point; value in actual measuring units

3	**71** **72**	**Pos. min** **Pos. max**	Minimum position value Maximum position value

These two values are the limits for the **input monitoring** within the programming mode of nominal values or for the direct entry (e.g. reference measure). Additionally, they fix the maximum positioning range if the function **software limit switch** is activated (refer to storage location 3/**73**)

condition: **Pos. min < Pos. max**

exception: Pos. min = Pos. max = 0

In this case, the input monitoring and software limit switch function are disabled.

An error message is issued if the entered nominal value is too small or too big (refer to section 4.11.1).

For rotary table positioning (refer to storage location 3/**8**), Pos. min/max are preset internally (Pos. min = 0, Pos. max = programmed counting range).

	XXXXXXXX	max. digits incl. sign (-) and decimal point; value in actual measuring units

3	**73**	**SWswitch**	Software limit switches

If the actual position is above or below the value programmed at storage locations 3/**71** and 3/**72**, the drive is stopped and the */fault* signal is output (High → Low change). The drive can then be moved only in the opposite direction (refer to section 4.11.1).

For rotary table positioning (refer to storage location 3/**8**), the limit switch function is deactivated internally; this also applies if »Pos. min« = »Pos. max« = 0 has been programmed (refer to storage location 3/**71**/**72**).

0	inactive	limit switch function is disabled
1	pos.	monitoring of the actual position during moving
2	⇑ start	as above, however: When starting a sentence, it is checked if the newly selected nominal position (e.g. nominal position of the previous sentence plus length of the new sentence) is still within the admissible range. If this is not the case, the start will not be effected and the limit switch will trigger

3	74	**HWswitch**	Hardware limit switches

Specifies at which data input, the signals of the limit switches are to be read in (refer to sections 4.11.2 and 4.10.1); meaning of the signal levels:

- Low level = limit switch has triggered (corresponds to an open input)
- High level = ready to operate

0	inactive	signals of hardware limit switches are ignored
1	input1.0	limit switch signals at the 1st data input E1, decade 10^0
2	input1.1	limit switch signals at the 1st data input E1, decade 10^1
3	input1.2	limit switch signals at the 1st data input E1, decade 10^2
4	input1.3	limit switch signals at the 1st data input E1, decade 10^3
5	input1.4	limit switch signals at the 1st data input E1, decade 10^4
6	input1.5	limit switch signals at the 1st data input E1, decade 10^5
7	input2.0	limit switch signals at the 2nd data input E2, decade 10^0
8	input2.1	limit switch signals at the 2nd data input E2, decade 10^1
9	input2.2	limit switch signals at the 2nd data input E2, decade 10^2
10	input2.3	limit switch signals at the 2nd data input E2, decade 10^3
11	input2.4	limit switch signals at the 2nd data input E2, decade 10^4
12	input2.5	limit switch signals at the 2nd data input E2, decade 10^5
13	input3.0	limit switch signals at the 3rd data input E3, decade 10^0
14	input3.1	limit switch signals at the 3rd data input E3, decade 10^1
15	input3.2	limit switch signals at the 3rd data input E3, decade 10^2
16	input3.3	limit switch signals at the 3rd data input E3, decade 10^3
17	input3.4	limit switch signals at the 3rd data input E3, decade 10^4
18	input3.5	limit switch signals at the 3rd data input E3, decade 10^5

GEL 8610 only:

19	input4.0	limit switch signals at the 4th data input E4, decade 10^0
20	input4.1	limit switch signals at the 4th data input E4, decade 10^1
21	input4.2	limit switch signals at the 4th data input E4, decade 10^2
22	input4.3	limit switch signals at the 4th data input E4, decade 10^3
23	input4.4	limit switch signals at the 4th data input E4, decade 10^4
24	input4.5	limit switch signals at the 4th data input E4, decade 10^5

3	**75**	**!Pos.in**	Ext. data input of a nominal position/length
Specifies at which data input the nominal position or length is to be applied (in BCD code, max. 6 digits), optionally with sign and length specifier; refer to section 4.10.1			
0		program	no external data specification
1		input1	data specification without sign at the 1st data input E_1 (range of values: 0 ... 999,999)
2		i1 +/-	data specification with sign at the 1st data input E_1 (range of values: –799,999 ... +799,999; sign: decade 10^5, bit 2^3; + ≙ Low)
3		i1 p/l	data specification with length specifier without sign at the 1st data input E_1 (range of values: 0 ... +799,999; length: decade 10^5, bit 2^3 = High)
4		i1 +- pl	data specification with length specifier and with sign at the 1st data input E_1 (range of values: –399,999 ... +399,999; length: decade 10^5, bit 2^3; sign: decade 10^5, bit 2^2)
5		input 2	data specification without sign at the 2nd data input E_2
6		i2 +/-	data specification with sign at the 2nd data input E_2
7		i2 p/l	data specification with length specifier without sign at the 2nd data input E_2
8		i2 +- pl	data specification with length specifier and with sign at the 2nd data input E_2
9		input 3	data specification without sign at the 3rd data input E_3
10		i3 +/-	data specification with sign at the 3rd data input E_3
11		i3 p/l	data specification with length specifier without sign at the 3rd data input E_3
12		i3 +- pl	data specification with length specifier and with sign at the 3rd data input E_3

GEL 8610 only:

13	input 4	data specification without sign at the 4th data input E_4
14	i4 +/-	data specification with sign at the 4th data input E_4
15	i4 p/l	data specification with length specifier without sign at the 4th data input E_4
16	i4 +- pl	data specification with length specifier and with sign at the 4th data input E_4

3	**76**	**Corr.in**	Ext. data input of a correction value
Specifies to which data input the correction value is to be applied (in BCD code, max. 6 digits), optionally with sign (refer to sections 4.7 and 4.10.1)			
0		program	no external data specification
1		input 1	data specification without sign at the 1st data input E1 (range of values: 0 ... 999,999)
2		i1 +/-	data specification with sign at the 1st data input E1 (range of values: –799,999 ... +799,999; sign: decade 10^5, bit 2^3; + ≙ Low)
3		input 2	data specification without sign at the 2nd data input E2
4		i2 +/-	data specification with sign at the 2nd data input E2
5		input 3	data specification without sign at the 3rd data input E3
6		i3 +/-	data specification with sign at the 3rd data input E3

GEL 8610 only:

7	input 4	data specification without sign at the 4th data input E4
8	i4 +/-	data specification with sign at the 4th data input E4

3	**77**	**Speed in**	Ext. data input of a speed value
Specifies to which data input the speed rate specification is to be applied (in BCD code, max. 6 digits), optionally with continuous sentence processing specifier '➧' (decade 10^5, bit 2^3 ≙ High); refer to sections 4.2, 4.3, and 4.10.1			
0		program	no external data specification
1		input 1	data specification <u>without</u> continuous sentence processing at the 1st data input E1 (range of values: 0 ... 999,999)
2		i1 ➧	data specification <u>with</u> continuous sentence processing at the 1st data input E1 (range of values: 0 ... +799,999)
3		input 2	data specification <u>without</u> continuous sentence processing at the 2nd data input E2
4		i2 ➧	data specification <u>with</u> continuous sentence processing at the 2nd data input E2
5		input 3	data specification <u>without</u> continuous sentence processing at the 3rd data input E3
6		i3 ➧	data specification <u>with</u> continuous sentence processing at the 3rd data input E3

GEL 8<u>6</u>10 only:

7	input 4	data specification <u>without</u> continuous sentence processing at the 4th data input E4
8	i4 ➧	data specification <u>with</u> continuous sentence processing at the 4th data input E4

3	**78** **79**	**Reserve**	
not used for the standard version			
		- - - - - - - -	

3	**80**	**!Pos.out**	Data output of nominal position values
	81	**=Pos.out**	Data output of actual position
	82	**Corr.out**	Data output of correction value

Specifies at which optional output module the data are to be output (in BCD code, max. 6 digits), optionally with sign and/or *data ready* (DR) signal (refer to section 4.10.2)

0	inactive	no data output
1	output 1	data without sign at the 1st data output (range of values: 0 ... 999,999)
2	o1 +/-	data with sign at the 1st data output (range of values: –799,999 ... +799,999; sign: decade 10^5, bit 2^3; + ≙ Low)
3	o1 DR	data with *data ready* signal without sign at the 1st data output (range of values: 0 ... +799,999; DR: decade 10^5, bit 2^3 ≙ High)
4	o1 +- DR	data with *data ready* signal and with sign at the 1st data output (range of values: –399,999 ... +399,999; DR: decade 10^5, bit 2^3; sign: decade 10^5, bit 2^2)
5	output 2	data without sign at the 2nd data output
6	o2 +/-	data with sign at the 2nd data output
7	o2 DR	data with *data ready* signal without sign at the 2nd data output
8	o2 +- DR	data with *data ready* signal and with sign at the 2nd data output
9	output 3	data without sign at the 3rd data output
10	o3 +/-	data with sign at the 3rd data output
11	o3 DR	data with *data ready* signal without sign at the 3rd data output
12	o3 +- DR	data with *data ready* signal and with sign at the 3rd data output

GEL 8610 only:

13	output 4	data without sign at the 4th data output
14	o4 +/-	data with sign at the 4th data output
15	o4 DR	data with *data ready* signal without sign at the 4th data output
16	o4 +- DR	data with *data ready* signal and with sign at the 4th data output

3	**83** **84**	**Reserve**	
not used for the standard version			
		- - - - - - - -	

3	**85**	**Ref.val2**	Value of the 2nd reference measure
Under the precondition that the *reference2/1* signal is active (High level at the corresponding terminal) this value is loaded into the actual position counter as soon as the drive exceeds the reference point; the value specified at this location can also be changed via direct entry in the Automatic mode (refer to storage location 3/**9** and section 4.6); 1st reference measure: 3/**15**			
		XXXXXXXX	max. 8 digits incl. sign (-) and decimal point; value in actual measuring units

3	**86**	**Reserve**	
• • •			*not used for the standard version*
3	**89**	**Reserve**	
		- - - - - - - -	

3	**90**	**DeltaS=0**	Zero Delta_s
Specifies whether the *zero Delta_s* signal shall be active for resetting a contouring error which has developed during the interrupted (stop) reset state (refer to section 4.5.3)			
0		inactive	signal is disabled
1		active	signal is enabled

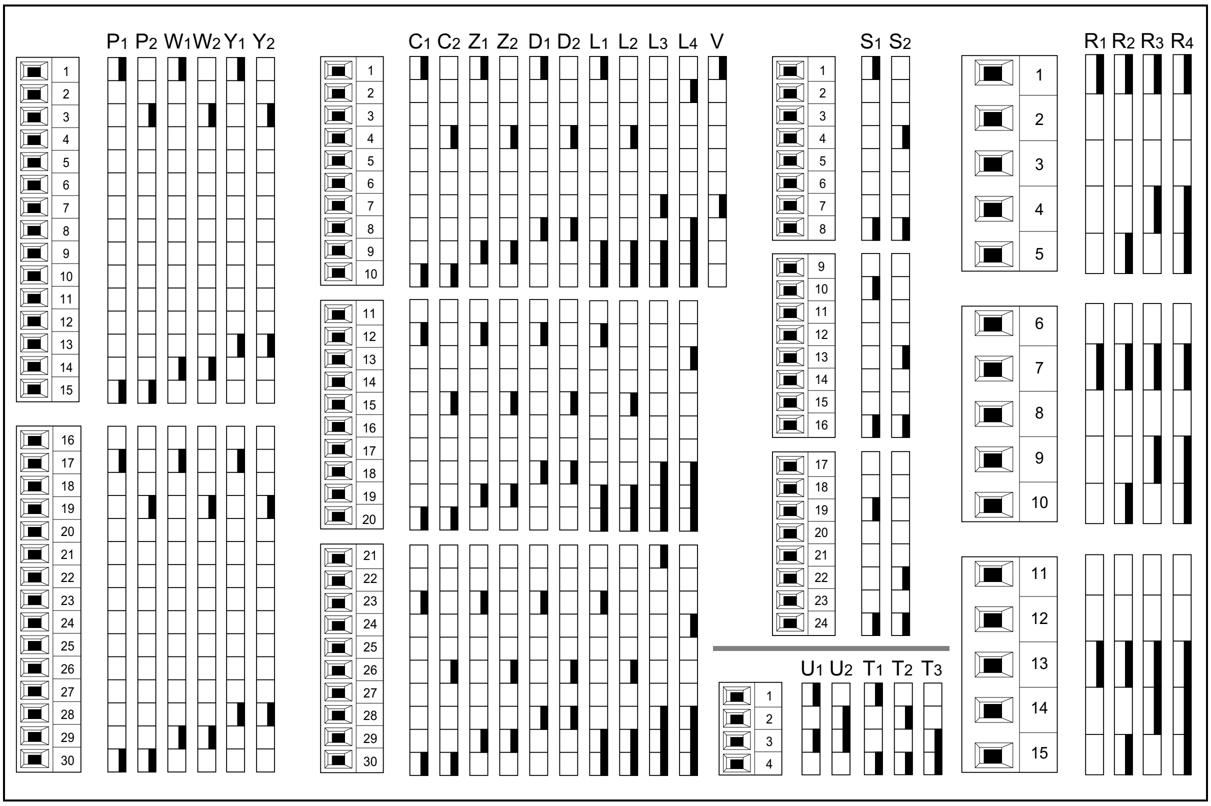
3	**91**	**PowerCal**	Calibration after powering-on
Specifies whether the axis is exempted from the calibration mode that is defined in system parameter 1/**2**			
0		asSystem	standard, i.e. no exemption
1		inactive	no calibration for this axis

Storage locations for machine parameters

Connector designations

terminal strip	data con-nector	module / function
	A	data output: logic outputs
C		1 to 3 count inputs for incremental encoders 5/24 V
D		1 to 3 analog outputs 0...±10 V
	E	data inputs (High level ≙ logic 1)
L		data output: power outputs
N		power supply
P		control inputs/outputs
R		data output: (relay) contact outputs
S		3 inputs for absolute encoders with serial data output (SSI)
	T	intelligent serial interface RS 422 for special protocols
	U	intelligent serial interface RS 485 for PROFIBUS applications
V	V	serial bus RS 422 / RS 485; serial interface V.24 / RS 232 C
W		input for absolute encoders (parallel, High level ≙ logic 1)
Y		input for absolute encoders (parallel, Low level ≙ logic 1)
Z		1 to 3 count inputs for incremental encoders 24 V

Terminal strip coding



X186003A

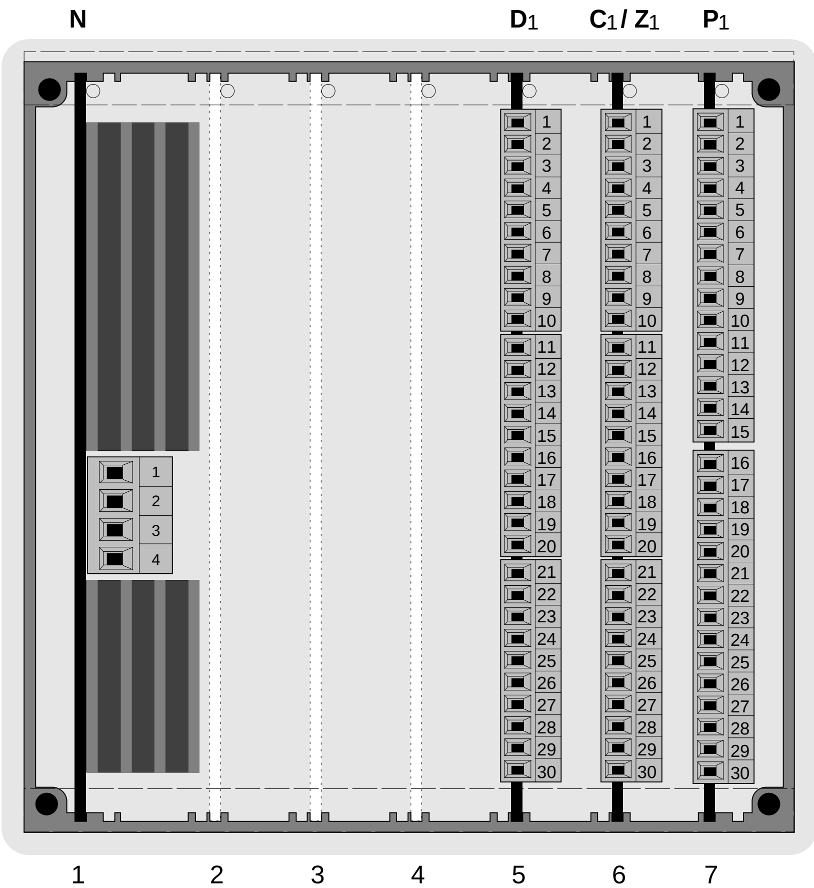
Modules which may exist several times (e.g. R) are marked by a corresponding index (1 to 4 at max.). A certain **address code** is allocated to each index. It is achieved via specific jumper positions on the module cards, which are indicated at the bottom of each connection diagram. The terminal sockets of the modules and the terminals strips supplied are coded and labelled accordingly,

e. g. R_2:

R	
	1
✗	2
	3
	4

Module arrangement of the GEL 8310 (back of the Controller)

Configuration example: positioning controller for 3 axes with incremental encoders, without additional modules (demonstrated for devices <u>without</u> CE certification)

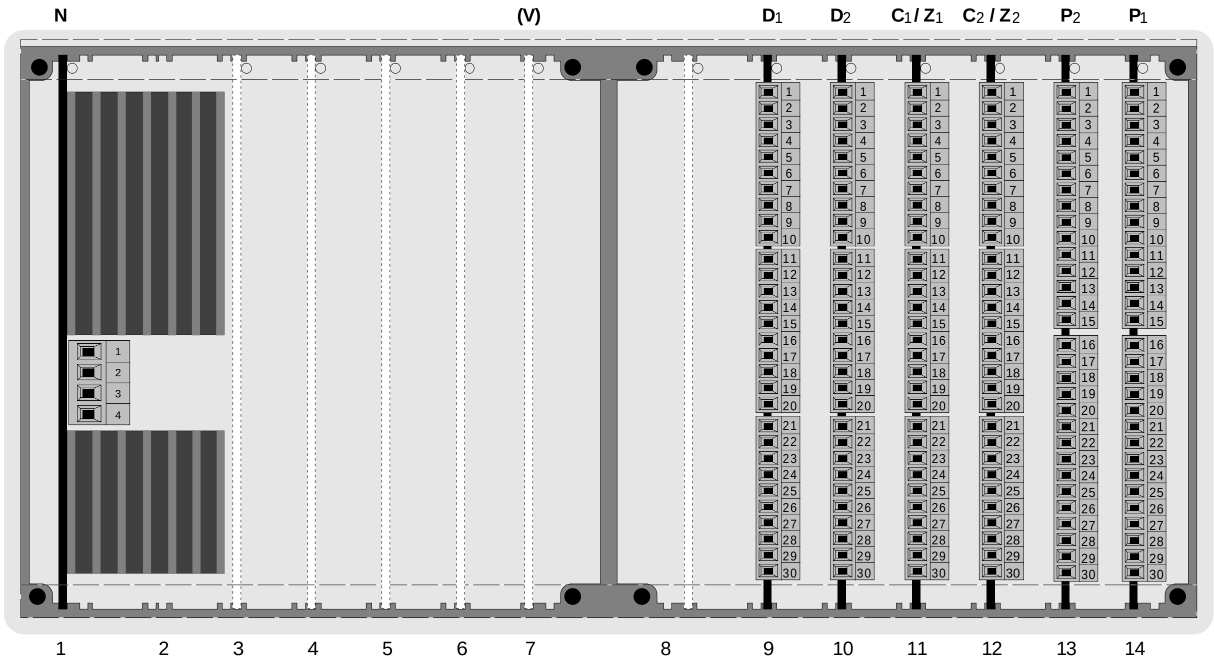


X1860003

slot	assign-ment	module	terminal strip, connector
1	fixed	power supply	N
2	variable	actual value inputs; analog outputs, data inputs/outputs; serial interface	C/Z, W, Y, S; D; E, A, L, R; T, U, V
3	variable	actual value inputs; analog outputs, data inputs/outputs; serial interface	C/Z, W, Y, S; D; E, A, L, R; T, U, V
4	variable	I actual value inputs; analog outputs, data inputs/outputs; serial interface	C/Z, W, Y, S; D; E, A, L, R; T, U, V
5	variable	actual value inputs; analog outputs, data inputs/outputs; serial interface	C/Z, W, Y, S; D; E, A, L, R; T, U, V
6	variable	actual value inputs; analog outputs, data inputs/outputs; serial interface	C/Z, W, Y, S; D; E, A, L, R; T, U, V
7	fixed	control inputs/outputs	P

Module arrangement of the GEL 8610 (back of the Controller)

Configuration example: positioning controller for 6 axes with incremental encoders, without additional modules (demonstrated for devices <u>without</u> CE certification)



X1860003

slot	assign-ment	module	terminal strip, connector
1 + 2	fixed	power supply	N
3…6	variable	actual value inputs; analog outputs, data inputs/outputs; serial interface	C/Z, W, Y, S; D; E, A, L, R; T, U, V
7	variable	actual value inputs; serial interface	C/Z, S; T, U, V
8…12	variable	actual value inputs; analog outputs, data inputs/outputs; serial interface	C/Z, W, Y, S; D; E, A, L, R; T, U, V
13	fixed	control inputs/outputs II	P_2
14	fixed	control inputs/outputs I	P_1

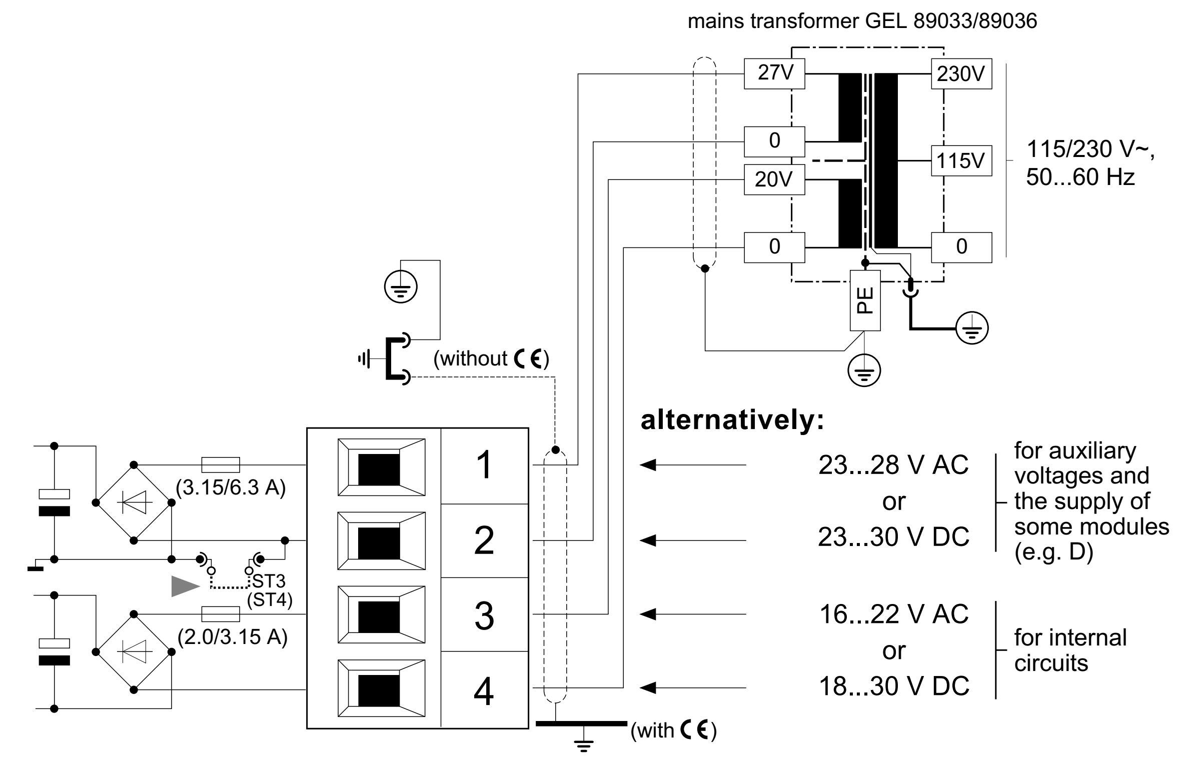
Connection diagrams

In the following diagrams, the **internal** circuitry of the module cards is sketched on the **left**. The **connections** to be effected, i.e. the signals, their direction, and other data, are shown on the **right**.

ATTENTION

With the P, E, A, W, Y modules you must determine by means of jumpers if the **internal (auxiliary) supply voltage** or an **external voltage** shall be used for the signals or the encoder supply.
For this reason, check the correct configuration of these jumpers relative to the desired application. Using a wrong assignment can cause the module being **demaged**! The jumpers are set to 'internal' (INT) at the factory.

N	Terminal strip	**Power supply** for internal circuits and for signals (auxiliary voltages)

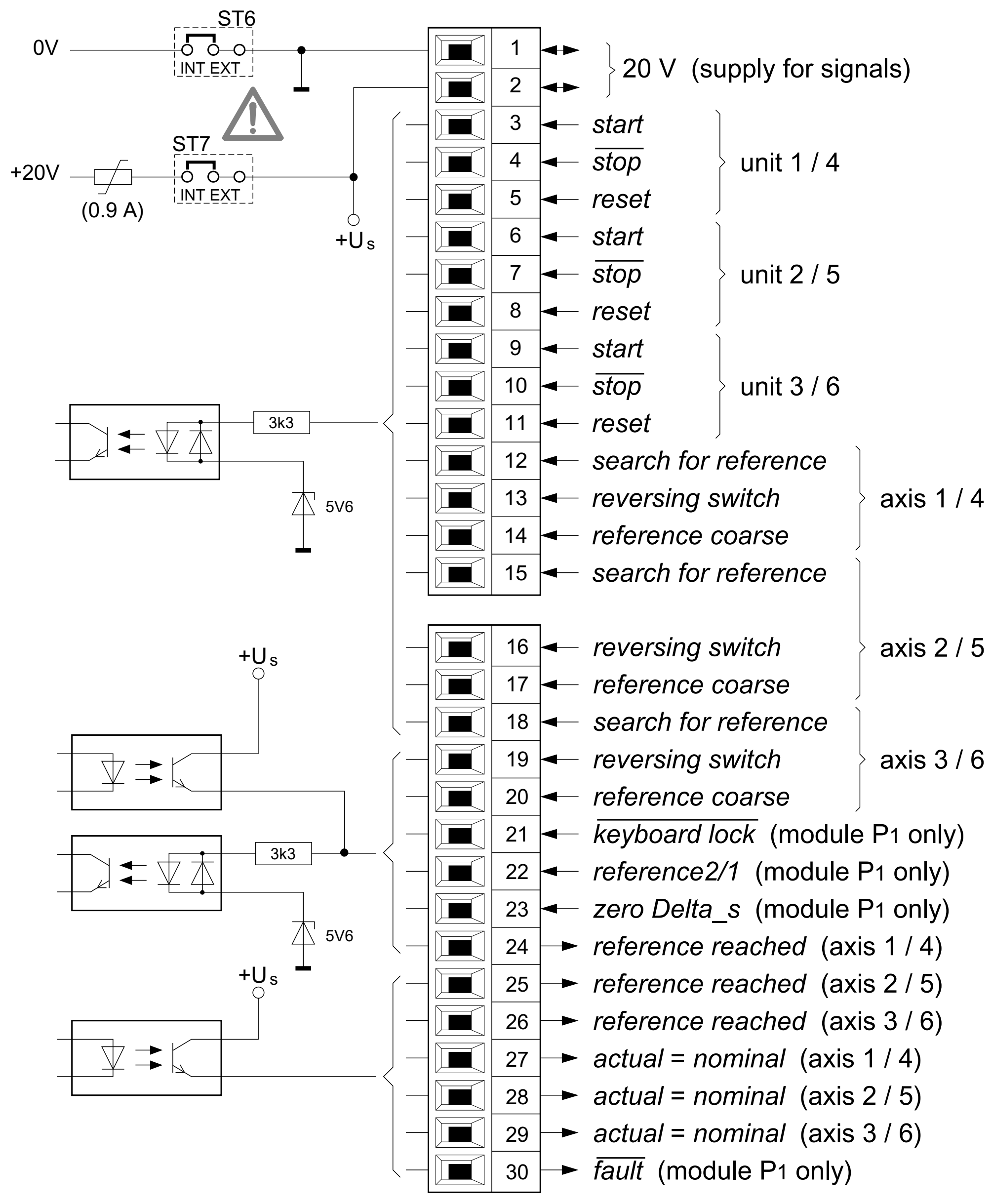


* With **DC** supply insert wire bridge in terminal strip ST3
(for devices without CE certification: ST4)

E180025I

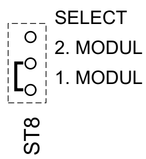
For the screening and earthing please refer to section 7.2, under 'EMC measures'.

P	Terminal strip	**Control inputs/outputs** with aux. voltage 20 V, I_{max} = 20 mA (per output)

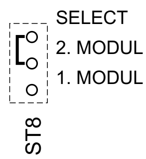


Address Coding

first module P1 (compulsory for GEL 83xx: unit/axis 1 ... 3)



second module P2 (for GEL 86xx only: unit/axis 4 ... 6)



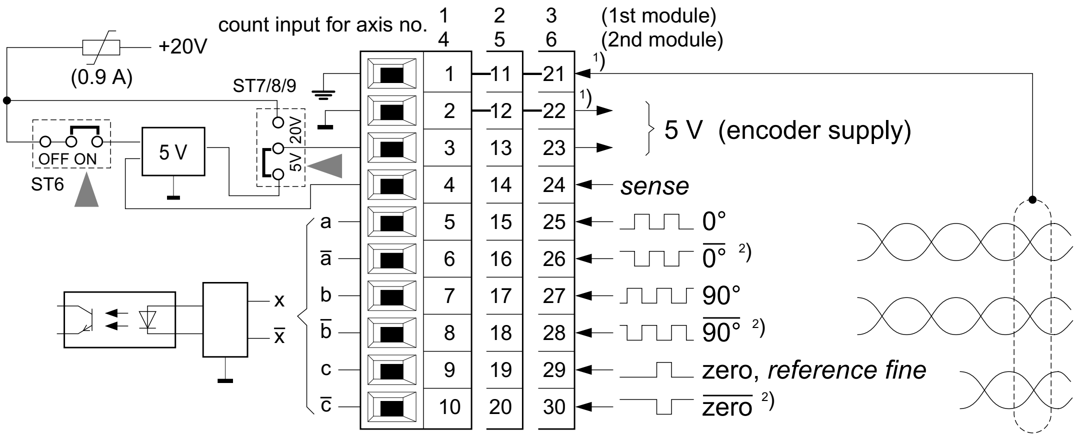
E180025J

For the Controllers without CE certification the EPROM containing the Controller software is located on module P1.

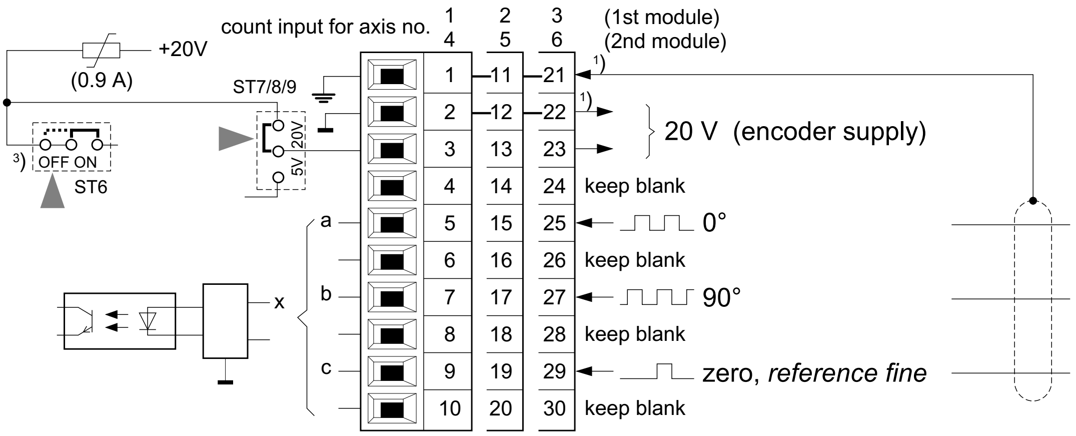
With some options the pin layout may be modified; refer to appendix O.

C, **Z**	Terminal strip	**Count input for 1 to 3 incremental encoders** with power supply for 5/24 V encoders (**C**), reversible, or for 24 V encoders only (**Z**)

5 V version (TTL level)



20 V version



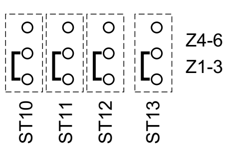
[1]) internal connections

[2]) without inverse signals: connect to ground for 5 V signal level, keep blank for 24 V signal level (this is the only possibilty to work with different signal levels for encoders and e.g. sensors at a single terminal strip)

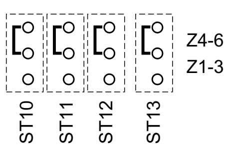
[3]) position "OFF" only if <u>none</u> of the 3 connectable encoders is operated at 5 V (applies for count module C only)

Address Coding

first count module C1 / Z1 (compulsory for GEL 83xx)



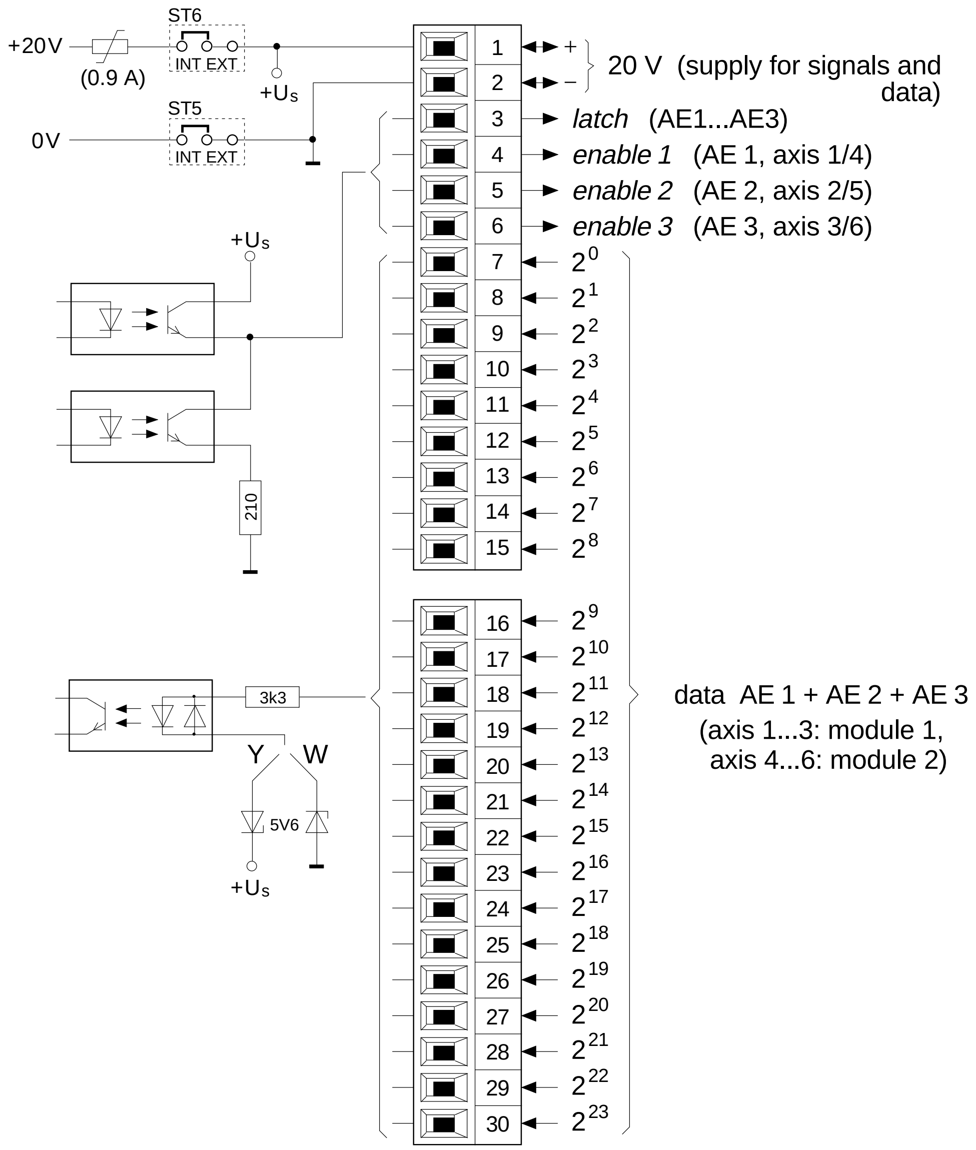
second count module C2 / Z2 (for GEL 86xx only)



E180025K

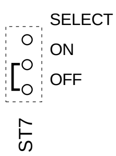
On the **C** module the encoder supply voltage is preset to **5 V** for all three inputs.

W, Y	Terminal strip	**Actual value input for 1 to 3 absolute encoders** with parallel data outputs, High-active (**W**) or Low-active (**Y**); with aux. voltage 20 V (power supply for encoder and data)

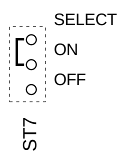


Address Coding

first module W_1/Y_1 (compulsory for GEL 83xx)



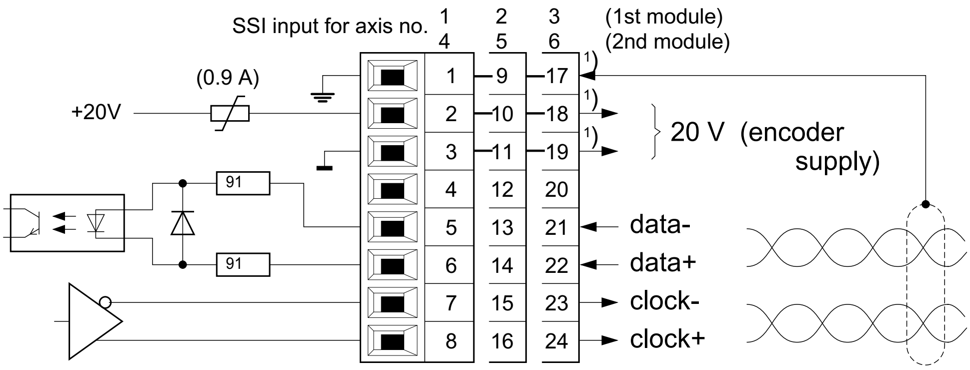
second module W_2/Y_2 (for GEL 86xx only)



E180025M

The encoders used must make available their data **within 100 µsec** after the signal *enable* has been given.

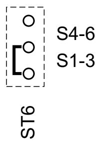
S	Terminal strip	**Actual value input for 1 to 3 SSI encoders** = absolute encoders wit synchronous serial data output; with aux. voltage 20 V (power supply for encoder)



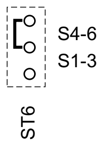
[1]) internal connections

Address Coding

first module S1
(compulsory for
GEL 83xx)

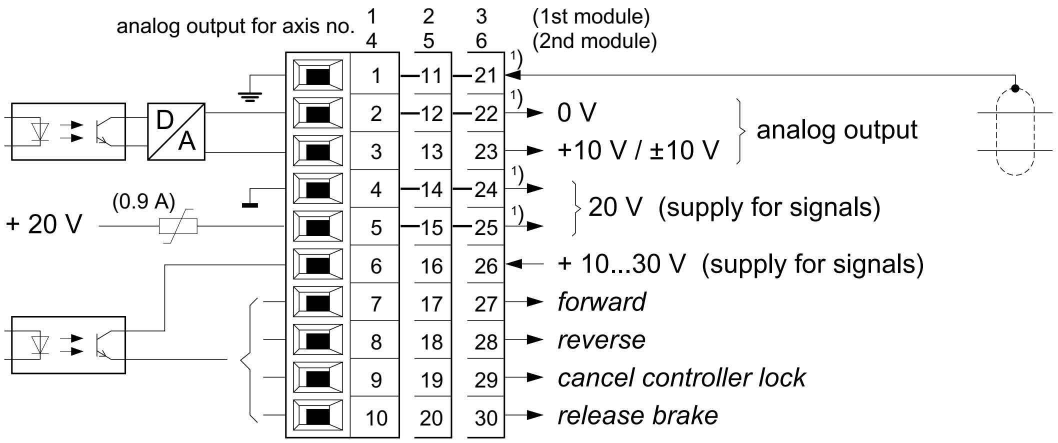


second module S2
(for GEL 86xx only)



E180025N

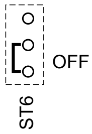
D	Terminal strip	**1 to 3 analog outputs** 0…±10 V/+10V, 10 mA, potential-free, 1.22 mV resolution, sustained short circuit-proof; with aux. voltage 20 V; control outputs: 20 mA at maximum



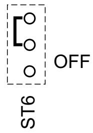
1) internal connections

Address Coding

first module D1 (compulsory for GEL 83xx)



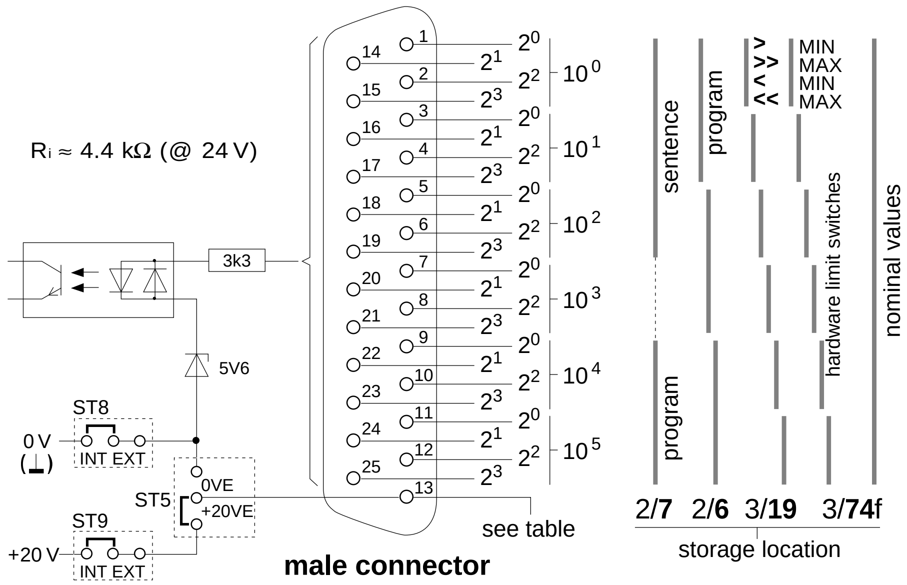
second module D2 (for GEL 86xx only)



E180025C

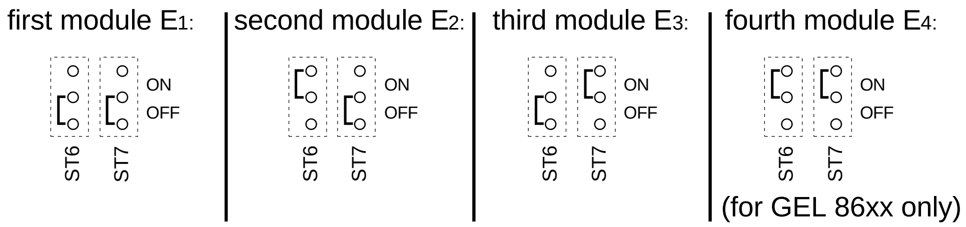
For the power supply of the module there must be a (auxiliary) voltage fed to terminals **1** and **2** of the **N** module.

E	Connector	**Data input** with auxiliary voltage 20 V



signal voltage	pin 13	ST5	ST8	ST9
internal 20 V (standard)	+ 20 V	+20VE	INT	INT
external U_S e. g. 24 V	- U_S (0 V)	0VE	EXT	EXT

Address Coding (data inputs/outputs)



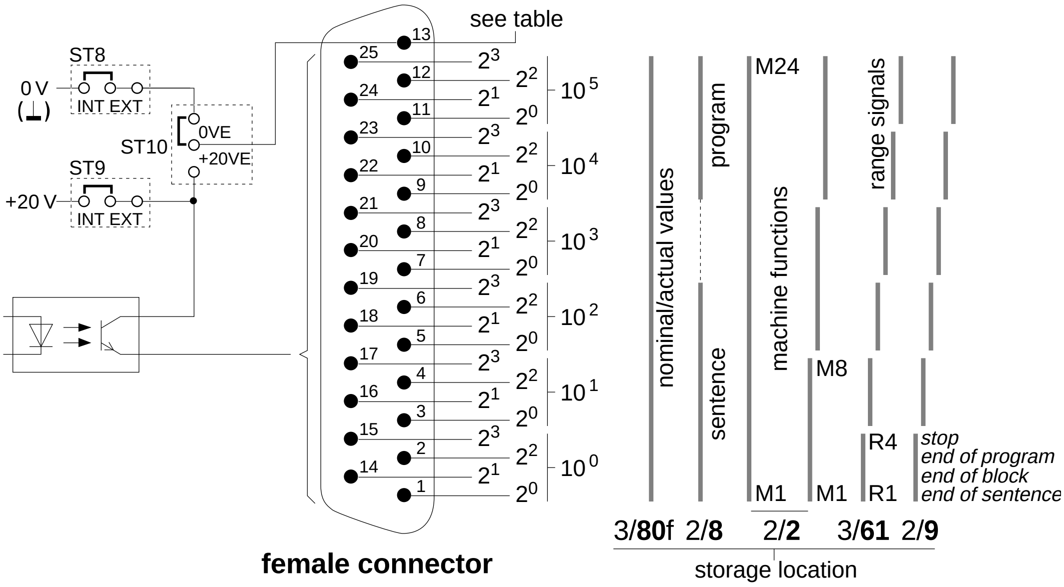
(With combined data input and output, address encoding applies to both.)

E180047A

The power supply for the signals ist preset to **internal** at the factory.

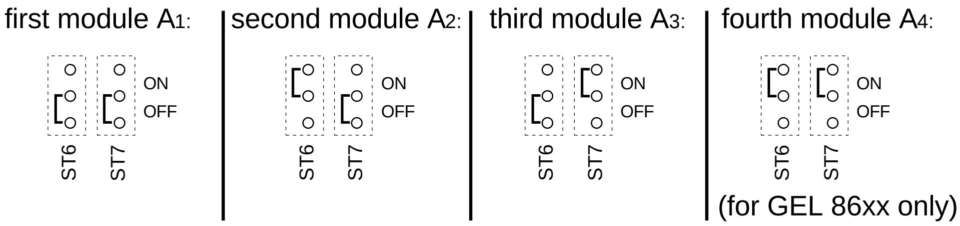
With combined data input/output (connectors E and A on a single module) this is valid for both input and output (common jumpers ST8 and ST9; refer to connector A on the next page). If you use **external** voltages for the signals you have to change all jumpers ST8 and ST9 as well as ST5 and ST10 according to both tables for connectors E and A. Otherwise the external voltage supply will be **short-circuited** causing **damages** on the module.

A	Connector	**Data output (logic outputs)** with aux. voltage 20 V, I_{max} = 10 mA (per output)



signal voltage	pin 13	ST10	ST8	ST9
internal 20 V (standard)	0 V	0VE	INT	INT
external Us e. g. 24 V	+Us	+20VE	EXT	EXT

Address Coding (data inputs/outputs)



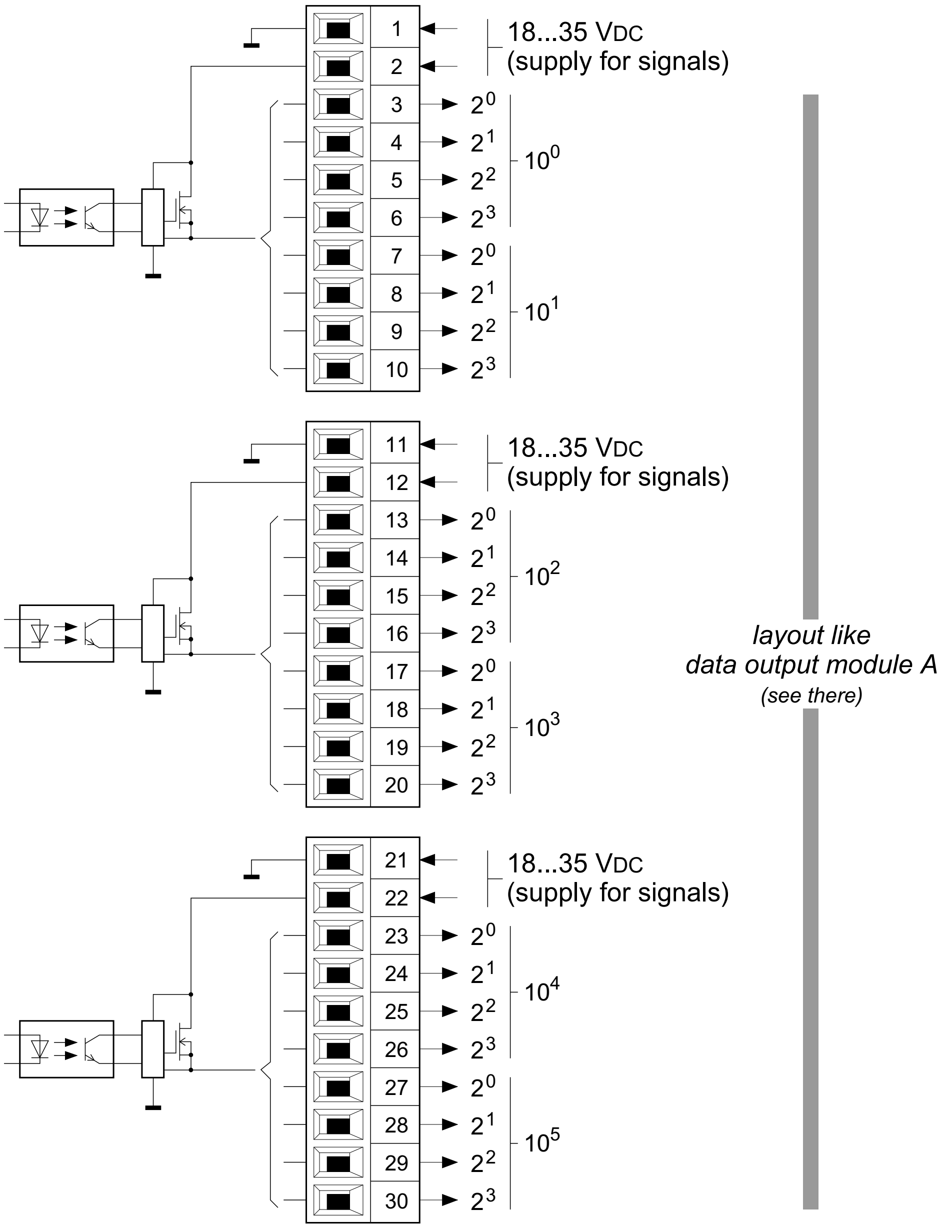
(With combined data input and output, address encoding applies to both.)

E180047D

The power supply for the signals ist preset to **internal** at the factory.

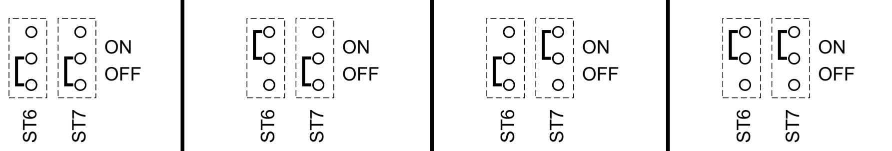
With combined data input/output (connectors E and A on a single module) this is valid for both input and output (common jumpers ST8 and ST9; refer to connector E on the previous page). If you use **external** voltages for the signals you have to change all jumpers ST8 and ST9 as well as ST5 and ST10 according to both tables for connectors E and A. Otherwise the external voltage supply will be **short-circuited** causing **damages** on the module.

L	Terminal strip	**Data output (power outputs)** High-active, sustained short circuit-proof, I_{max} = 500 mA per output



Address Coding (data inputs/outputs)

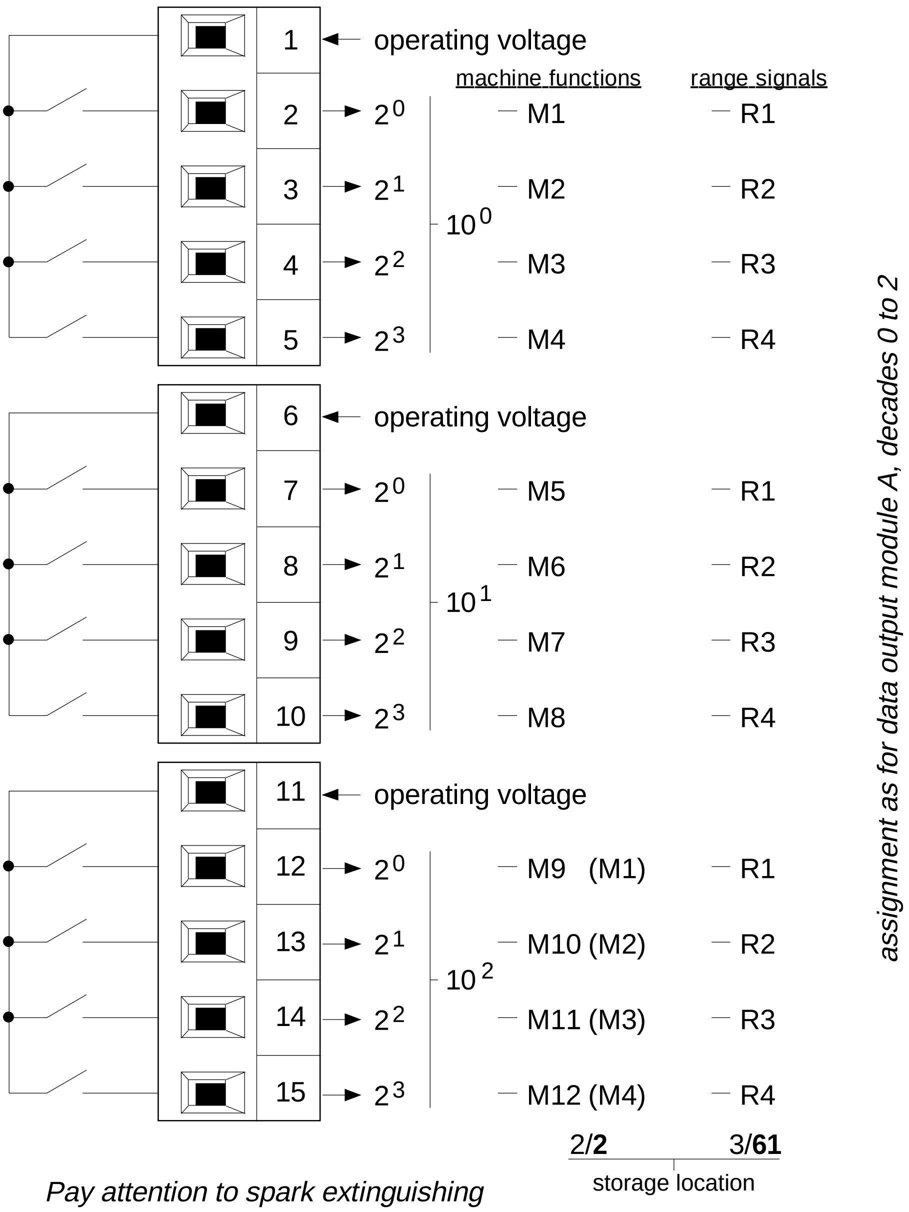
first module L1: | second module L2: | third module L3: | fourth module L4:



(GEL 86xx only)

E180025O

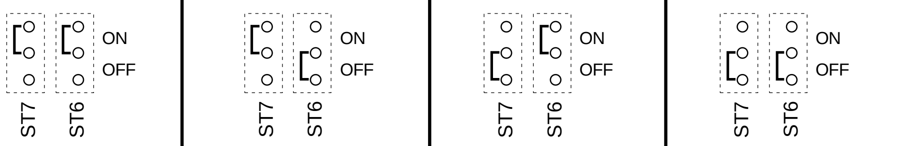
R	Terminal strip	**Data output (contact outputs)** contacts 240 V~/1 A at maximum



Pay attention to spark extinguishing for relay contacts!

Address Coding (data inputs/outputs)

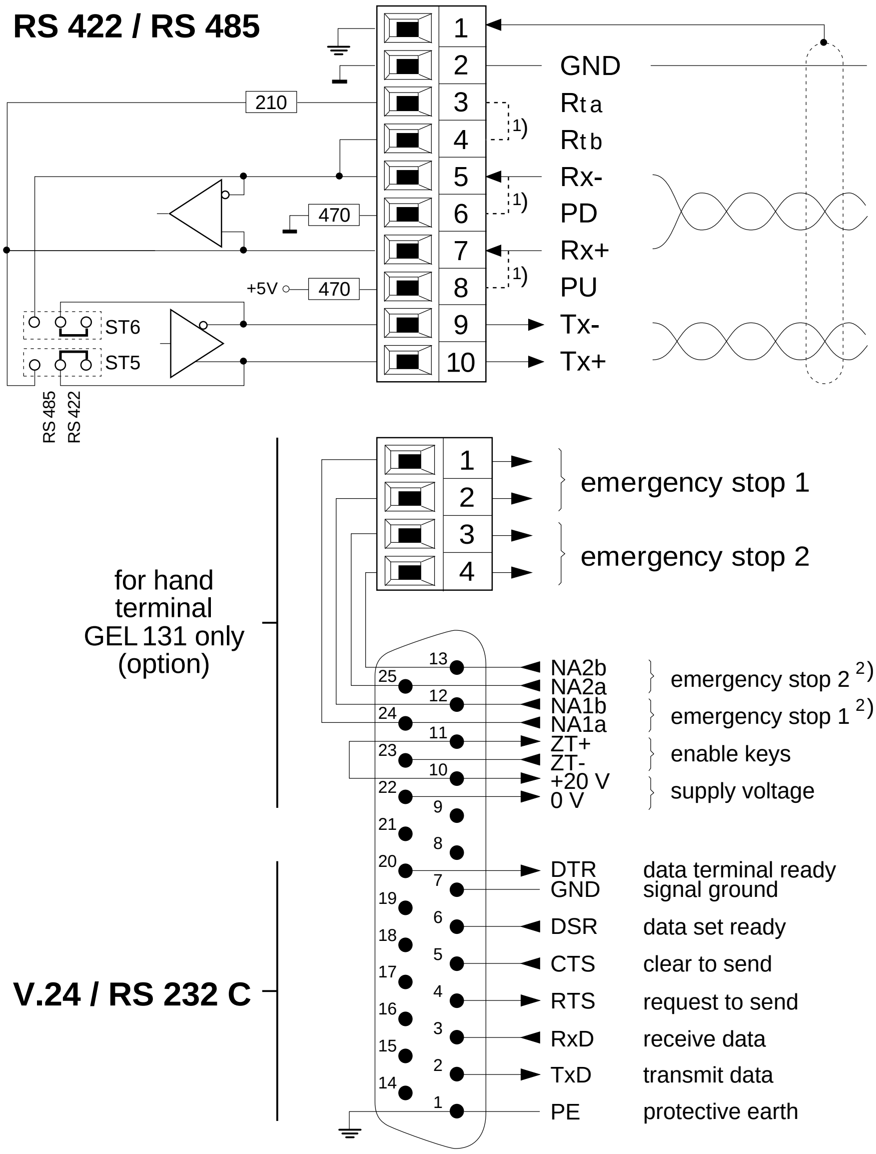
first module R1: | second module R2: | third module R3: | fourth module R4:



(for GEL 86xx only)

E180025L

V	Terminal strip Connector	**Serial bus RS 422 / RS 485** **Serial interface RS 232 C / V.24**



female connector

1) If transmission problems occur insert bridges for terminal resistance (Rt) as well as for pull-up and pull-down resistance (PU, PD) at the terminal unit.

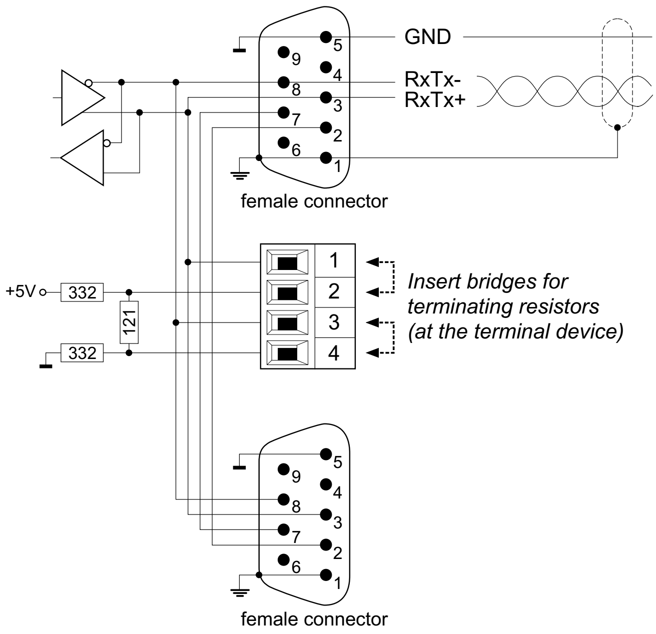
2) Signals are not processed internally.

E180047B

For the power supply of the module – and the optional hand terminal GEL 131 – there must be a (auxiliary) voltage fed to terminals **1** and **2** of the **N** module.

If the module is inserted the description of the Op1 option in appendix O will give you further information (additionally, Op2 for the GEL 131).

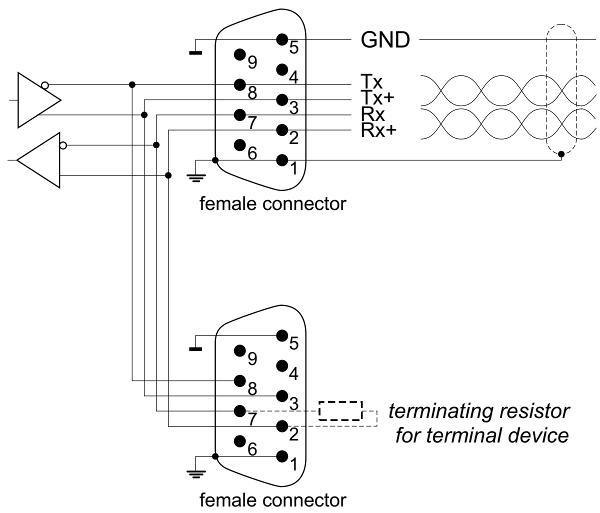
U	Connector	**Intelligent interface RS 485** for PROFIBUS applications



E180047C

If the module is inserted the description of the Op3 option in appendix O will give you further information.

T	Connector	**Intelligent interface RS 422** for special protocols



E180047F

A separate description is supplied if the module is inserted.

For the power supply of the modules there must be a (auxiliary) voltage fed to terminals **1** and **2** of the **N** module.

Pin Layout

Operational data

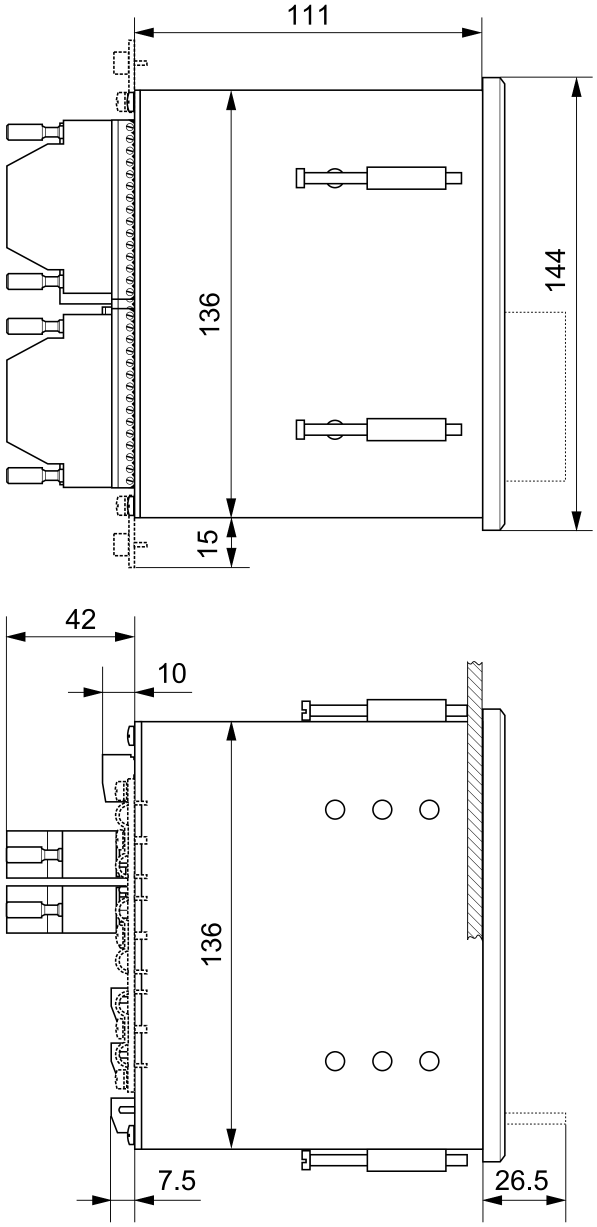
Power requirements	
– for the Controller circuits	18 ... 30 VDC or 16 ... 22 VAC
GEL 8310	current consumption max. 1.2 A (slow-blow miniature fuse 2.0 A)
GEL 8610	current consumption max. 2.0 A (slow-blow miniature fuse 3.15 A)
– for signal circuits, encoder supply, and some modules	23 ... 30 VDC or 23 ... 28 VAC
GEL 8310	max. 2.4 A (slow-blow miniature fuse 3.15 A)
GEL 8610	max. 4.0 A (slow-blow miniature fuse 6.3 A)
– for use of mains transformer GEL 8903*x* (design according to VDE 0551)	115/230 VAC 50/60 Hz (also refer to section 7.2)
GEL 8310	power consumption max. 130 VA
GEL 8610	power consumption max. 170 VA
Count inputs (terminal strips C and Z)	opto-decoupled
logic level	Low (24 V): 0 ... +5 V, High (24 V): +15 ... +30 V Low (5 V): 0 ... +0.8 V High (5 V): +2.5 ... +5 V
input resistance	approx. 3.4 kΩ at 20 V, approx. 0.4 kΩ at 5 V
input frequency	≤ 100 kHz, pulse width of the zero signal ≥ 5 µsec
encoder supply	20 V or 5 V (switched from 20 V supply), 0.9 A
Actual value inputs (terminal strip W/Y)	opto-decoupled
data	response on *enable* signal max. 100 µsec
encoder/signal supply	20 V, 0.9 A
SSI inputs (terminal strip S)	opto-decoupled
data, clock	according to RS 422 specification
clock frequency	200 kHz
encoder supply	20 V, 0.9 A

Digital inputs	opto-decoupled
logic level	Low: 0 ... +5 V, High: +15 ... +30 V
input resistance	approx. 5 kΩ (at 20 V)
auxiliary voltage	20 V, 0.9 A
Digital outputs	opto-decoupled, NPN transistor, open Emitter
I_{max}	20 mA, data output A: 10 mA
auxiliary voltage	20 V, 0.9 A
max. output voltage	30 V
Power outputs (terminal strip L)	opto-decoupled
I_{max}	500 mA per output; short circuit-proof (higher loads will cause oscillating)
external power supply	18…35 VDC
Contact outputs (terminal strip R)	240 VAC / 1 A (spark extinguishing is recommended)
Analog outputs (terminal strip D)	potential-free
voltage range	-10.000 V ... 0 ... +10.000 V
resolution	1.22 mV (14-bit D/A converter)
I_{max}	10 mA, sustained short circuit-proof
max. offset error	± 0.7 mV related to 23 °C
offset temperature coefficient	typ. 0.20 mV / 10 K, max. 1.00 mV / 10 K
Counting range	-2^{31} ... $+2^{31}-1$
Display range	–9 999 999 ... 99 999 999
Number of axes	
GEL 8310	max. 3 (1 ... 3 units)
GEL 8610	max. 6 (1 ... 6 units)
Storage locations for nominal values	
GEL 8310	7168
GEL 8610	6416
partition	max. 99 programs per unit max. 999 sentences per program

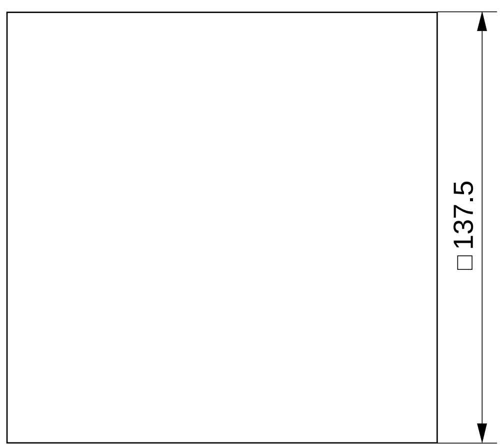
Control scan time	typ. 1 msec per connected axis
Power failure-safe storing	EEPROM, service life: 10,000 write operations per byte or 10 years
Climatic application class	KWF (according to DIN 40040 specification)
relative humidity	up to 95%, no condensation
operating temperature range	0 °C ... 50 °C
storage temperature range	-20 °C ... +80 °C
EMC (by observing the set-up instructions)	*for Controllers with CE certification only*
noise emission	according to EN 50081-2 (1994)
noise immunity	according to EN 50082-2 (1995)
Display height	
– display A	14 mm, seven segments
– displays B, C	5 mm, dot matrix 5x7
Max. line cross-section	
– terminal strips C/Z, P, …	1.5 mm^2 (3.8 mm grid)
– terminal strip N	2.5 mm^2 (5.1 mm grid)
– terminal strip R	2.5 mm^2 (7.6 mm grid)
Weight (incl. all components)	
GEL 8310	approx. 1.5 kg
GEL 8610	approx. 3 kg
Protective class	
– front without slot for memory card	IP 50
– front with slot for memory card	IP 20

Dimensions

Controller GEL 8310



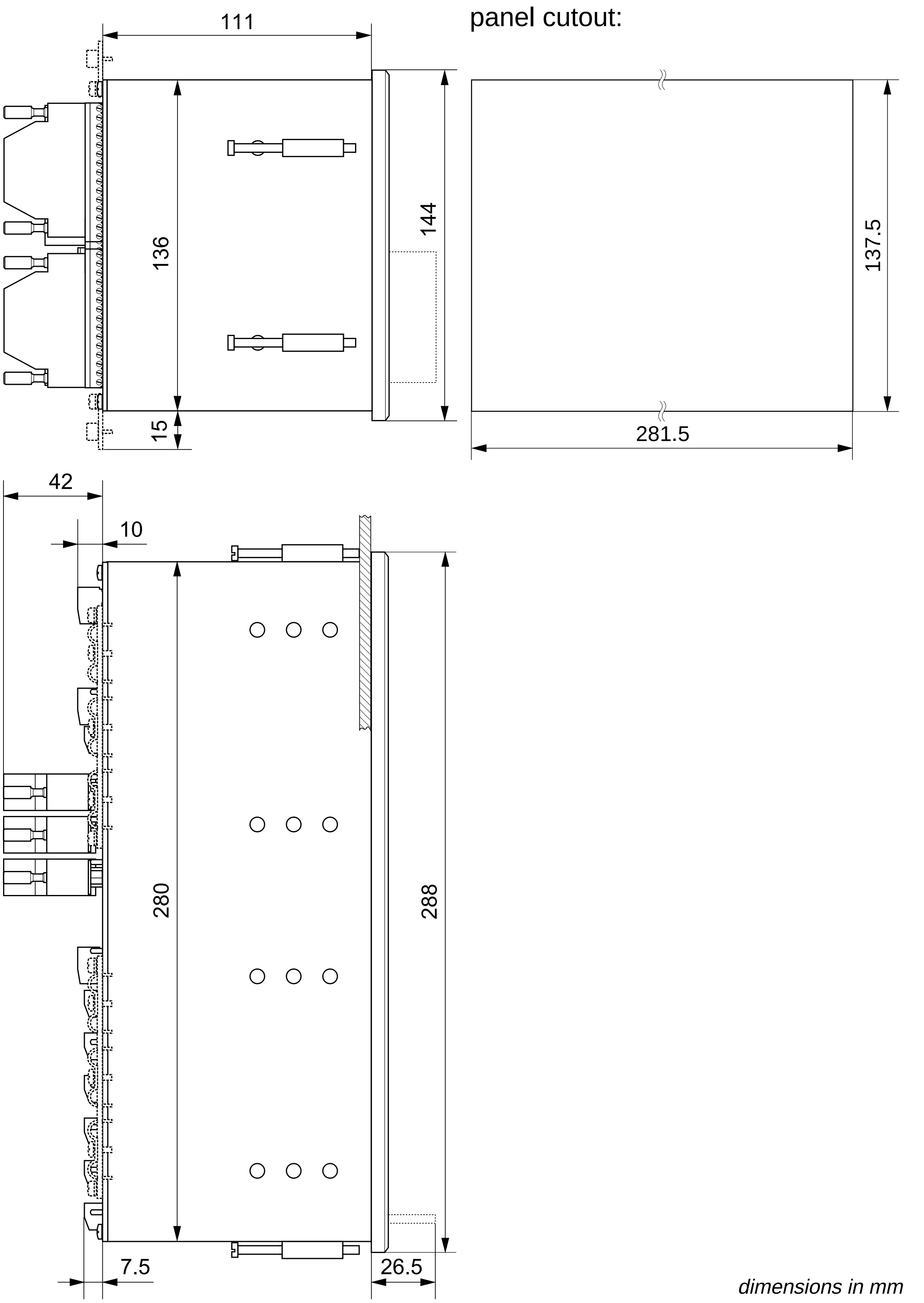
panel cutout:



dimensions in mm

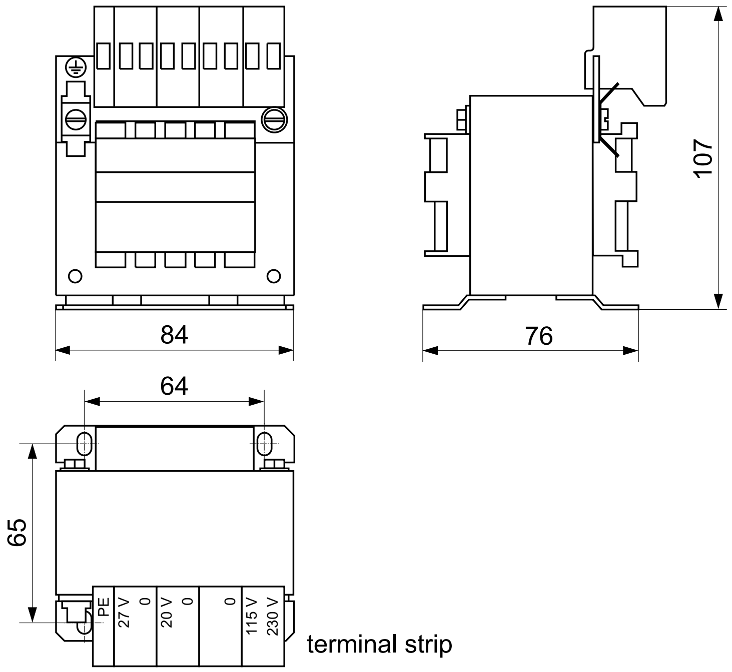
X/D183036D..F

Controller GEL 8610



X/D186036D..F

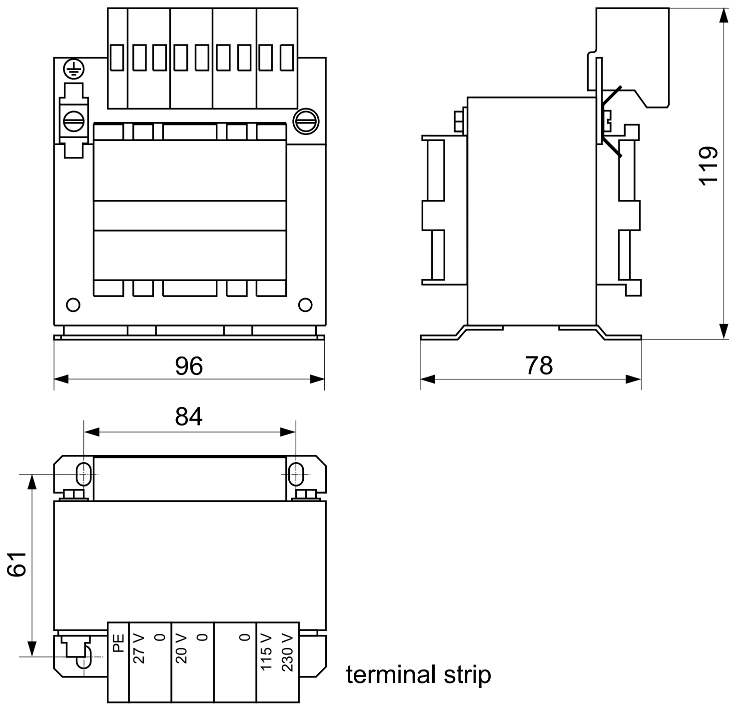
GEL 89033 mains transformer



dimensions in mm

E183036G

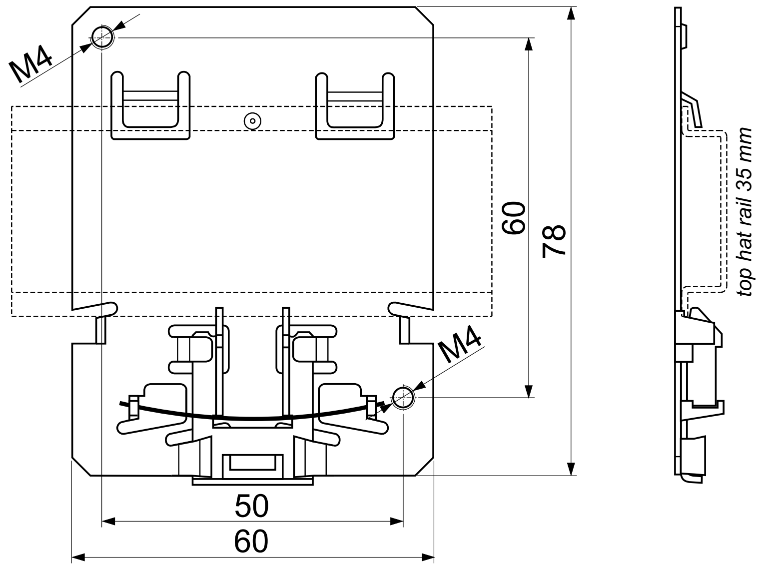
GEL 89036 mains transformer



dimensions in mm

E186036G

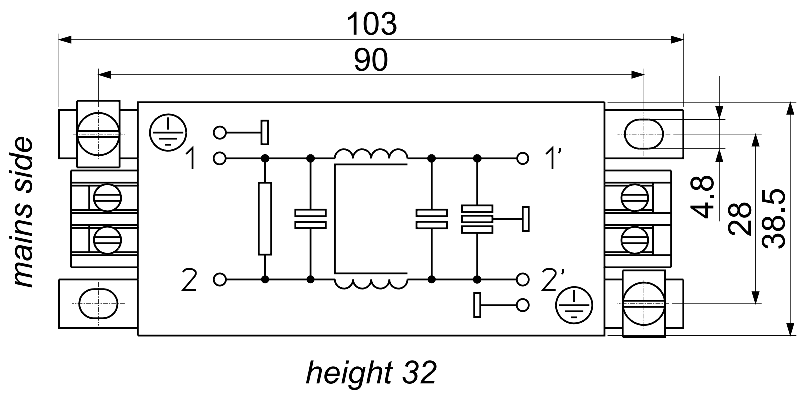
GEL 7922 clamping plate for GEL 89033/36 mains transformer



dimensions in mm

E6101CH

GEL 7925 mains suppression filter for mains-based voltage spikes > 2.5 kV, 250 V≅, 50/60 Hz, 2 A



dimensions in mm

E6101AH

Types and coding

Format: G E L

	8	X	1	0	X	X	X	X	X	X	X	X	X	X
place	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Every "X" from the 7th place in the coding stands for exactly **one** module to be specified; the total number of the optional plug-in modules (from the 6th place) is limited to **5** (GEL 8310) or **10** (GEL 8610). "0" in the type code means that no choice has to be made for this place.

Basic device

8	**3**	**1**	**0**	X	X	X	X	0	X	X	X	0	0
1	2	3	4										

Controller enclosed in a **144**x144 mm casing for up to **3** axes with positioning software, incl. power supply (module N)

8	**6**	**1**	**0**	X	X	X	X	X	X	X	X	X	X
1	2	3	4										

Controller enclosed in a **288**x144 mm casing for up to **6** axes with positioning software, incl. power supply (module N)

control inputs/-outputs

(compulsory)

8	3	1	0	**X**	X	X	X	0	X	X	X	0	0
8	6	1	0	**X**	X	X	X	X	X	X	X	X	X
				5									

X	module(s)	
	without memory card option	
A	for 1 axis	(*P1*)
B	for 2 axes	(*P1*)
C	for 3 axes	(*P1*)
D	for 4 axes	(*P1*+*P2*)
E	for 5 axes	(*P1*+*P2*)
F	for 6 axes	(*P1*+*P2*)
	with memory card option	
G	for 1 axis	(*P1*)
H	for 2 axes	(*P1*)
J	for 3 axes	(*P1*)
K	for 4 axes	(*P1*+*P2*)
L	for 5 axes	(*P1*+*P2*)
M	for 6 axes	(*P1*+*P2*)

Analog outputs

8	3	1	0	X	**X**	X	X	0	X	X	X	0	0
8	6	1	0	X	**X**	X	X	X	X	X	X	X	X
					6								

X	outputs (module *D1* / *D2*)		
0	—	(–/–)	
A	1	(1 / –)	
B	2	(2 / –)	
C	3	(3 / –)	
D	4	(3 / 1)	GEL 8610 only
E	5	(3 / 2)	GEL 8610 only
F	6	(3 / 3)	GEL 8610 only
G	3	(2 / 1)	GEL 8610 only
H	4	(2 / 2)	GEL 8610 only
J	5	(2 / 3)	GEL 8610 only
K	2	(1 / 1)	GEL 8610 only
L	3	(1 / 2)	GEL 8610 only
M	4	(1 / 3)	GEL 8610 only

For example, ‘K’ means that there is one analog output for axis no. 1 and one for axis no. 4.

Actual value inputs

8	3	1	0	X	X	**X**	**X**	0	X	X	X	0	0
8	6	1	0	X	X	**X**	**X**	**X**	X	X	X	X	X
						7	8	9					

X	module	
A	1 count input 24 V	*Z*
B	2 count inputs 24 V	*Z*
C	3 count inputs 24 V	*Z*
D	1 count input 5 V or 24 V	*C*
E	2 count inputs 5 V or 24 V	*C*
F	3 count inputs 5 V or 24 V	*C*
N	input for 1…3 absolute encoders with Low-active outputs (multiplexed)	*Y*
P	input for 1…3 absolute encoders with High-active outputs (multiplexed)	*W*
S	3 inputs for absolute encoders with synchronous-serial output (SSI)	*S*

Data inputs/-outputs

8	3	1	0	X	X	X	X	0	**X**	**X**	**X**	0	0
8	6	1	0	X	X	X	X	X	**X**	**X**	**X**	**X**	**X**
									10	11	12	13	14

X	module	
0	—	
1	data input/output, inputs High-active	*E/A*
3	data output	*A*
5	data input, inputs High-active	*E*
6	serial interface RS232C/V.24 and serial bus RS422/RS485 with real-time clock (<u>once</u> possible only)	*V*
7	as 6, but <u>without</u> real-time clock	
8	intelligent interface RS422 for special protocols	*T*
9	intelligent interface RS485 for PROFIBUS applications (GEL 8910.0103 option required)	*U*
B	as 7; additionally, PC software BB8110 included (single-user licence)	
C	as 6; additionally, PC software BB8110 included (single-user licence)	
L	power outputs	*L*
R	contact (relay) outputs	*R*
W	as 7; additionally, PC software LB-Flex for WINDOWS (PG 8012) included, German → Controller type GEL 8x1**9**	
V	as W, but English	

<u>Example</u>: GEL 8310 J C F00 6RR00

Positioning controller enclosed in a small casing (8310) with the facility of storing data on a memory card and controlling 3 axes (J); 3 analog outputs (C); 1 module with 3 count inputs for incremental encoders 5 V / 24 V (F); 1 serial interface V.24 with real-time clock (6) and 2 modules with contact outputs (R)

Accessories

designation	order no.
Mains transformer 115/230 V, 50...60 Hz sec.: 20 V, 25 VA and 27 V, 70 VA	GEL 89033
Mains transformer 115/230 V, 50...60 Hz sec.: 20 V, 35 VA and 27 V, 110 VA	GEL 89036
Clamping plate for mains transformer GEL 89033	GEL 7922
Mains suppression filter with current compensated chokes, 250 VAC	GEL 7925
Memory card (EEPROM)	GEL 89001
EPROM set for software updates	GEL 89005
Hand terminal	GEL 131

Options

designation	documentation	order no.
• **Communication**		
LB2 procedure (inserted as standard)	DS11-89100101	GEL 8910.0**1**01
GEL 131 procedure (incl. LB2)	DS11-89100102	GEL 8910.0**1**02
PROFIBUS incl. LB2 (module U required)	DS11-89100103	GEL 8910.0**1**03
as before; additionally, GEL 131 included	DS11-89100103	GEL 8910.0**1**04
• **Functions**		
Circular interpolation	DS11-89100201	GEL 8910.0**2**01
Synchro control	DS11-89100202	GEL 8910.0**2**02
Compact single-axis device	DS11-89100204	GEL 8910.0**2**04
Flying saw	DS11-89100205	GEL 8910.0**2**05
Rotating knife	DS11-89100207	GEL 8910.0**2**07

Note: The options within a group cannot be combined, i.e., only one option GEL 8910.x**1**xx and/or GEL 8910.x**2**xx can be inserted at a time. Options belonging to different groups can, however, be combined.
Descriptions concerning **inserted** options you will find in appendix O.

Specifications

Possible options

Op1	LB2 procedure
Op2	GEL 131 procedure
Op3	PROFIBUS
Op10	Circular interpolation
Op20	Synchro control
Op40	Compact single-axis device
Op50	Flying saw
Op70	Rotating knife
Op__	_________________

Key Reference

for the

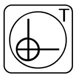
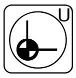
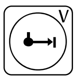
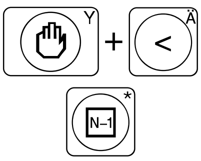
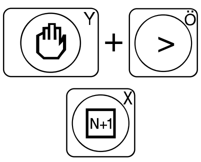
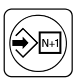
Positioning Controllers

GEL 8310 / 8610

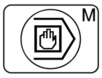
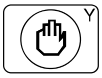
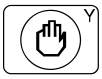
(valid from standard software version 14.00)

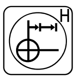
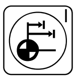
Lenord + Bauer

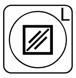
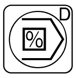
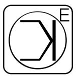
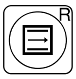
 Subject to change without notice

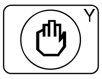
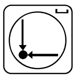
GEL 8x10 keys	GEL 8610 keys	function	m*
F1	—	select next variant	M
F1 + ·		display date/time (option; without F1 when displaying fault messages)	A
F1 + 0	Error	display stored fault messages	A
F1 + 1	 —	direct input of a reference measure absolute coordinates offset	A N
F1 + 2	 —	search for reference relative coordinates offset	A N
F1 + 3		direct input of a correction value	A
F1 + 4	—	direct input of a parking position	A
F1 + 9	Teach In	activate teach-in mode	N
F1 + ⬆		slow speed reverse go to the beginning of the previous sentence	A N
F1 + ⬇		slow speed forward go to the beginning of the next sentence	A N
F1 + ⇨		storing and changing to the next nominal value with the same type	N

* valid mode: A = **A**utomatic mode
N = Programming mode for **n**ominal values
M = Programming mode for **m**achine parameters

GEL 8x10 keys	GEL 8610 keys	function	m
F2 + 0	 Auto (S)	activate the programming mode for nominal values return to Automatic mode	A N, M
F2 + 1	—	activate the programming mode for machine parameters	A
F2 + 8	Copy (C)	copy axis parameters	M
F2 + ⬆	 + << (Z)	fast speed reverse	A
F2 + ⬇	 + >> (Ü)	fast speed forward	A
F2 + ➔		write to optional memory card	A

GEL 8x10 keys	GEL 8610 keys	function	m
F3	—	select previous variant	M
F3 + ·	—	display the version of built-in software	A
F3 + +/−	—	display signal states	A
F3 + 0	Unit (N)   	select the unit for display process. of relat. dimens. (position → length) process. of absol. dimens. (length → position) continuous sentence processing ON/OFF	A N N N
F3 + 1	— —	select program set program flow instruction »CALL Pr.«	A N
F3 + 2	—	set program flow instruction »JUMP Pr.«	N

GEL 8x10 keys	GEL 8610 keys	function	m
F3 + 3	—	set program flow instruction »JMP sent«	N
F3 + 4	—	set program flow instruction »IF I/O«	N
F3 + 7		clear memory or delete unit/program	N, M
F3 + 8	Copy (C)	copy sentences	N
F3 + 9	Def. Prog. End (K)	define end of program	N
F3 + 		delete sentence	N
F3 + ↑	Axis−1 (P) 	 go to the beginning of program	A N
F3 + ↓	Axis+1 (O) 	increment axis no. within the unit go to the end of program	A N
F3 + 	 	read optional memory card insert sentence	A N

GEL 8x10 keys	GEL 8610 keys	function	m
—	 + 	output *actual=nominal* signal manually	A

Configuration of the data input/output modules

Module: input **E** ☐ output **A** ☐ **L** ☐ **R** ☐ no. 1 ☐ 2 ☐ 3 ☐ 4 ☐ *)

de-cade	pin resp. terminal	function	unit/ axis
0	E/A: 1-14-2-15 L: 3-4-5-6 R: 2-3-4-5		
1	E/A: 3-16-4-17 L/R: 7-8-9-10		
2	E/A: 5-18-6-19 L: 13-14-15-16 R: 12-13-14-15		
3	E/A: 7-20-8-21 L: 17-18-19-20		
4	E/A: 9-22-10-23 L: 23-24-25-26		
5	E/A: 11-24-12-25 L: 27-28-29-30		

Module: input **E** ☐ output **A** ☐ **L** ☐ **R** ☐ no. 1 ☐ 2 ☐ 3 ☐ 4 ☐ *)

de-cade	pin resp. terminal	function	unit/ axis
0	E/A: 1-14-2-15 L: 3-4-5-6 R: 2-3-4-5		
1	E/A: 3-16-4-17 L/R: 7-8-9-10		
2	E/A: 5-18-6-19 L: 13-14-15-16 R: 12-13-14-15		
3	E/A: 7-20-8-21 L: 17-18-19-20		
4	E/A: 9-22-10-23 L: 23-24-25-26		
5	E/A: 11-24-12-25 L: 27-28-29-30		

*) Please cross what is applicable

Module: input **E** ☐ output **A** ☐ **L** ☐ **R** ☐ no. 1 ☐ 2 ☐ 3 ☐ 4 ☐ *)

de-cade	pin resp. terminal	function	unit/ axis
0	E/A: 1-14-2-15 L: 3-4-5-6 R: 2-3-4-5		
1	E/A: 3-16-4-17 L/R: 7-8-9-10		
2	E/A: 5-18-6-19 L: 13-14-15-16 R: 12-13-14-15		
3	E/A: 7-20-8-21 L: 17-18-19-20		
4	E/A: 9-22-10-23 L: 23-24-25-26		
5	E/A: 11-24-12-25 L: 27-28-29-30		

Module: input **E** ☐ output **A** ☐ **L** ☐ **R** ☐ no. 1 ☐ 2 ☐ 3 ☐ 4 ☐ *)

de-cade	pin resp. terminal	function	unit/ axis
0	E/A: 1-14-2-15 L: 3-4-5-6 R: 2-3-4-5		
1	E/A: 3-16-4-17 L/R: 7-8-9-10		
2	E/A: 5-18-6-19 L: 13-14-15-16 R: 12-13-14-15		
3	E/A: 7-20-8-21 L: 17-18-19-20		
4	E/A: 9-22-10-23 L: 23-24-25-26		
5	E/A: 11-24-12-25 L: 27-28-29-30		

*) Please cross what is applicable

Configuration of the storage locations

customer:	type: GEL 8___________________

system parameters

1		6		11		16		21		26	
2		7		12		17		22		27	
3		8		13		18		23		28	
4		9		14		19		24		29	
5		10		15		20		25		30	

unit parameters

	unit 1	unit 2	unit 3	unit 4	unit 5	unit 6
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						

axis parameters

	axis ____				
1		41		81	
2		42		82	
3		43		83	
4		44		84	
5		45		85	
6		46		86	
7		47		87	
8		48		88	
9		49		89	
10		50		90	
11		51		91	
12		52		92	
13		53		93	
14		54		94	
15		55		95	
16		56		96	
17		57		97	
18		58		98	
19		59		99	
20		60		100	
21		61		101	
22		62		102	
23		63		103	
24		64		104	
25		65		105	
26		66		106	
27		67		107	
28		68		108	
29		69		109	
30		70		110	
31		71		111	
32		72		112	
33		73		113	
34		74		114	
35		75		115	
36		76		116	
37		77		117	
38		78		118	
39		79		119	
40		80		120	

	axis ____				
1		41		81	
2		42		82	
3		43		83	
4		44		84	
5		45		85	
6		46		86	
7		47		87	
8		48		88	
9		49		89	
10		50		90	
11		51		91	
12		52		92	
13		53		93	
14		54		94	
15		55		95	
16		56		96	
17		57		97	
18		58		98	
19		59		99	
20		60		100	
21		61		101	
22		62		102	
23		63		103	
24		64		104	
25		65		105	
26		66		106	
27		67		107	
28		68		108	
29		69		109	
30		70		110	
31		71		111	
32		72		112	
33		73		113	
34		74		114	
35		75		115	
36		76		116	
37		77		117	
38		78		118	
39		79		119	
40		80		120	

axis parameters

	axis ____				
1		41		81	
2		42		82	
3		43		83	
4		44		84	
5		45		85	
6		46		86	
7		47		87	
8		48		88	
9		49		89	
10		50		90	
11		51		91	
12		52		92	
13		53		93	
14		54		94	
15		55		95	
16		56		96	
17		57		97	
18		58		98	
19		59		99	
20		60		100	
21		61		101	
22		62		102	
23		63		103	
24		64		104	
25		65		105	
26		66		106	
27		67		107	
28		68		108	
29		69		109	
30		70		110	
31		71		111	
32		72		112	
33		73		113	
34		74		114	
35		75		115	
36		76		116	
37		77		117	
38		78		118	
39		79		119	
40		80		120	

	axis ____				
1		41		81	
2		42		82	
3		43		83	
4		44		84	
5		45		85	
6		46		86	
7		47		87	
8		48		88	
9		49		89	
10		50		90	
11		51		91	
12		52		92	
13		53		93	
14		54		94	
15		55		95	
16		56		96	
17		57		97	
18		58		98	
19		59		99	
20		60		100	
21		61		101	
22		62		102	
23		63		103	
24		64		104	
25		65		105	
26		66		106	
27		67		107	
28		68		108	
29		69		109	
30		70		110	
31		71		111	
32		72		112	
33		73		113	
34		74		114	
35		75		115	
36		76		116	
37		77		117	
38		78		118	
39		79		119	
40		80		120	

axis parameters

	axis ____				
1		41		81	
2		42		82	
3		43		83	
4		44		84	
5		45		85	
6		46		86	
7		47		87	
8		48		88	
9		49		89	
10		50		90	
11		51		91	
12		52		92	
13		53		93	
14		54		94	
15		55		95	
16		56		96	
17		57		97	
18		58		98	
19		59		99	
20		60		100	
21		61		101	
22		62		102	
23		63		103	
24		64		104	
25		65		105	
26		66		106	
27		67		107	
28		68		108	
29		69		109	
30		70		110	
31		71		111	
32		72		112	
33		73		113	
34		74		114	
35		75		115	
36		76		116	
37		77		117	
38		78		118	
39		79		119	
40		80		120	

	axis ____				
1		41		81	
2		42		82	
3		43		83	
4		44		84	
5		45		85	
6		46		86	
7		47		87	
8		48		88	
9		49		89	
10		50		90	
11		51		91	
12		52		92	
13		53		93	
14		54		94	
15		55		95	
16		56		96	
17		57		97	
18		58		98	
19		59		99	
20		60		100	
21		61		101	
22		62		102	
23		63		103	
24		64		104	
25		65		105	
26		66		106	
27		67		107	
28		68		108	
29		69		109	
30		70		110	
31		71		111	
32		72		112	
33		73		113	
34		74		114	
35		75		115	
36		76		116	
37		77		117	
38		78		118	
39		79		119	
40		80		120	

Nominal value programs

customer:	type: GEL 8___________________

unit ___

prog.	sent.	type[1]	value

unit ___

prog.	sent.	type	value

[1] *e. g. P1 = position axis 1, L3 = length axis 3, PN = piece number, M = machine functions, S = speed, Sf = floating sentence processing, CALL Pr. etc.*

unit ___

prog.	sent.	type	value

unit ___

prog.	sent.	type	value