

Manual AP22 Analog Display



- 8 Digit display in DIN-enclosure 96 X 48 mm
- Supply voltage 10 – 35V DC
- All in- and outputs optically isolated
- Programmable input for +/-10V or +/-20mA
- RS232
- 2 Digital inputs and 4 digital outputs
- 12 Programmable cams / 4 outputs
- 24 Programmable nominal values
- 12 Programmable Cams over 4 outputs (cycle time 250 μ S)
- Programmable Linearization function
- Programmable PeakHold function

Table of contents

1	Introduction	4
1.1	GENERAL	4
1.2	IMPORTANT INFORMATION	5
1.3	EMC	5
1.4	DEFINITIONS	5
1.4.1	<i>Display units AWE</i>	5
1.4.2	<i>Parameter number</i>	5
1.4.3	<i>Notation</i>	6
1.4.4	<i>Edges</i>	6
2	Operation	7
2.1	KEY FUNCTIONS	7
2.2	KEY FUNCTIONS IN PROGRAMMING MODE	8
2.3	DISPLAY FUNCTIONS	9
2.3.1	<i>Status functions</i>	9
2.3.2	<i>Error messages</i>	9
2.3.3	<i>Survey of error messages</i>	10
3	Programming	11
3.1	AUTOMATIC MODE	11
3.1.1	<i>Monitor function</i>	12
3.1.2	<i>Displaying the type number</i>	13
3.1.3	<i>Displaying the software-version</i>	13
3.1.4	<i>Status in- and outputs</i>	14
3.2	CHANGING NOMINAL VALUES	15
3.3	CHANGING PARAMETERS	16
3.3.1	<i>Menus</i>	16
3.3.2	<i>Input parameters</i>	17
4	Functions	18
4.1	ANALOG INPUT	18
4.1.1	<i>Voltage input</i>	19
4.1.2	<i>Current input</i>	20
4.2	VELOCITY MEASUREMENT	21
4.3	JUSTAGE	22
4.3.1	<i>Preset using the parameter "Adjustment absolute value"</i>	22
4.3.2	<i>Using the input (input 1 or 2)</i>	22
4.3.3	<i>Using service parameter P[207]</i>	23
4.3.4	<i>Justage using front keys</i>	23
4.4	ASCII PROTOCOL	24
4.4.1	<i>Overview functions</i>	24
4.4.2	<i>General</i>	25
4.4.3	<i>Functions</i>	27
4.4.4	<i>Error messages</i>	33
4.5	CAMS	34
4.5.1	<i>General</i>	34
4.5.2	<i>Switching type (1) Cam with start- and end-value</i>	35
4.5.3	<i>Switching type (2) Greater than or equal to limit value</i>	35
4.5.4	<i>Switching type (3) Smaller than or equal to limit value</i>	35
4.5.5	<i>Switching type (4) Greater than or equal to limit value (Timer)</i>	36

4.5.6	Switching type (5) Smaller than or equal to limit value (Timer)	36
4.5.7	Dynamic cams	37
4.5.8	Start/stop cam	38
4.5.9	Output "Cams active"	38
4.6	LINEARIZATION	39
4.6.1	mode 0 "4-Quadrant"	40
4.6.2	mode 1 "Mirror Y-axis"	41
4.6.3	mode 2 "Mirror XY-axis"	41
4.7	PEAKHOLD FUNCTION	42
4.7.1	general.....	42
4.7.2	PeakHold 1.....	42
4.7.3	PeakHold 2.....	42
5	Parameters.....	43
5.1	MENU 1 CONFIG	43
5.2	MENU 2 ACTUAL	45
5.3	MENU 3 SERIAL	47
5.4	MENU 4 INPUT	48
5.5	MENU 5 OUTPUT.....	49
5.5.1	Submenu 5.1 – 5.4 Op1...4.....	49
5.6	MENU 6 CAM.....	50
5.6.1	Submenu 6.1 ... 6.12 CA1...12.....	50
5.7	MENU 8 LINEARIZATION.....	51
5.7.1	Submenu 8.1 Config.....	51
5.7.2	Submenu 11.2 ... 11.31 P1...P30.....	52
5.8	MENU 9 PEAKHOLD	53
5.8.1	Submenu 9.1 PeakHold 1	53
5.8.2	Submenu 9.2 PeakHold 2.....	53
5.9	OVERVIEW PARAMETERS.....	54
6	Connections.....	56
6.1	OVERVIEW CLAMP CONNECTIONS.....	56
6.2	SUPPLY.....	57
6.3	ANALOG INPUT	58
6.4	DIGITAL INPUTS.....	58
6.5	DIGITAL OUTPUTS.....	59
6.6	RS232	59
7	Technical specifications.....	60
7.1	SPECIFICATIONS	60
7.2	TYPEKEY	62
7.3	DIMENSIONS AP22	63
7.4	DIMENSIONS EMC BRACKET TYPE EMC-B02.....	64
7.4.1	Connections RS232 on 9P sub-D connector	65
7.5	DIMENSIONS PROTECTIVE HOOD TYPE CDS-B02.....	66

1 Introduction

1.1 General

The microcontroller based AP22 has a analog input for voltage or current signals. The 16bit input has a span of +/-10V or +/-20mA for all the known signals to be configured.

It is standard equipped with 2 digital inputs and 4 digital outputs, all free programmable. The AP22 features a RS232 communication port and a analog output.

With the 12 programmable cams several functions can be realized like limit detection.

Another feature of the AP22 is the memory for the nominal values. The 24 programmable values can be used to determine the positions of the cams.

The microcontroller reads the actual sensor value and calculates the display-value and the actual velocity. A programmable power failure protection makes sure that the actual value is stored in an EEPROM. The AP22 is configurable for numerous applications by adjusting its parameters.

The AP22 can be programmed with the DST2 PC-based software.

1.2 Important information

- ✓ The AP22 is a high-tech electronic product. To ensure safety and a correct functioning of the product it is important that only qualified specialists will install and operate the AP22.
- ✓ If through a failure or fault of the AP22 an endangering of persons or damage to plant is possible, this must be prevented using additional safety measures. These must remain operational in all possible modes of the AP22.
- ✓ Necessary repairs to the AP22 are only to be carried out by the manufacturer.

1.3 EMC

To ensure the best possible electromagnetic compatibility, it is recommended to pay attention to shielding and grounding the AP22:

- ✓ Shielding on both sides and with the largest possible contact area.
- ✓ Keep wiring as short as possible.
- ✓ Earth-connections should be short and with the highest possible wiring-diameter.
- ✓ Signal-cables and supply-cables must be separated.
- ✓ The EMC-bracket type EMC-B02 should be used.

1.4 Definitions

1.4.1 Display units AWE

The display units, referred to as AWE, is the value shown on the display without regarding the decimal point. The decimal point is only used for the comfort of the operator, but has no functional meaning.
(display = 347.4 >> AWE = 3474)

1.4.2 Parameter number

A parameter number is always shown in the format P[xxx]. It is possible that a parameter number appears in more than one menu.

1.4.3 Notation

Values can be displayed in different notations like binary or hexadecimal. The character behind the value shows in which notation the value is represented:

100D	<u>D</u> ecimal
238H	<u>H</u> exadecimal
244G	<u>G</u> ray
10010011B	<u>B</u> inary

for example 220D = DCH = 11011100B

1.4.4 Edges

L→H : rising edge (low to high)
H →L : falling edge (high to low)

2 Operation

2.1 Key functions



[P] key

- Cycle through monitoring displays
- Activate programming mode (in combination with other keys)



[+1] key

- View type number



[Cursor] key

- View software version
- View custom software version (in combination with the [Enter] key)



[Enter] key

- View status of inputs and outputs

2.2 Key functions in programming mode



[P] key

- one step back in menu
- discontinue programming mode
- discontinue changing nominal values/parameters (edit mode)
- LED is on when programming mode is active



[+1] key

- cycle through menu
- increase nominal value- / parameter number
- increase digit (in edit mode)



[Cursor] key

- activate edit mode
- move one digit to the left (in edit mode)

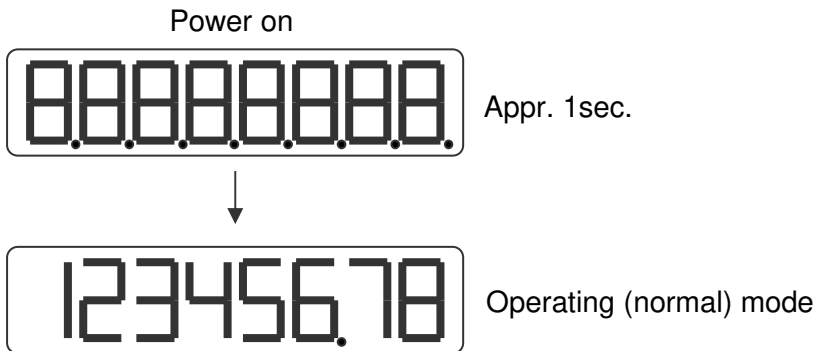


[Enter] key

- enter a submenu or parameter
- increase nominal value-/parameter number
- store a changed value
- clear value, hold down [Cursor] button (edit mode)

2.3 Display functions

2.3.1 Status functions

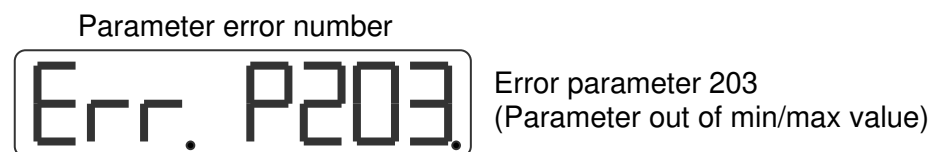


2.3.2 Error messages

There are two groups of errors:

- Parameter errors (error numbers 0...499, preceded by a “P”)
- General error (error numbers from 500 and up)

Example:



2.3.3 Survey of error messages

Error no:

000...499 Parameter error is displayed as PXXX on the display.

- 716 = Umin \geq Umax (analog output)
- 717 = Imin \geq Imax (analog output)
- 720 = Changelock nominal values active
- 721 = Changelock parameters active
- 732 = Function input-2 not valid (equal to input-1)

Error messages linearization (xx = P1 ...P30)

- 9xx = Linearization $X_n \leq X_{n-1}$
- 940 = Linearization (mode 1 or 2) $X_1 < 0$
- 941 = Linearization mode 1) $Y_1 < 0$

Error messages for cams (latst 2 digits = cam number)

- 1001...1012 length cam = 0 (cam begin = cam end) or
cam begin \leq cam end (with no counting range active)
- 1101...1112 length cam \leq Hysteresis
- 1201...1212 (2 * hysteresis) + cam length \geq counting range
- 1301...1312 cam begin and/or cam end outside counting range
(incl. hysteresis)
- 1401...1412 hysteresis too large ($>$ half counting range) only with timer
cams

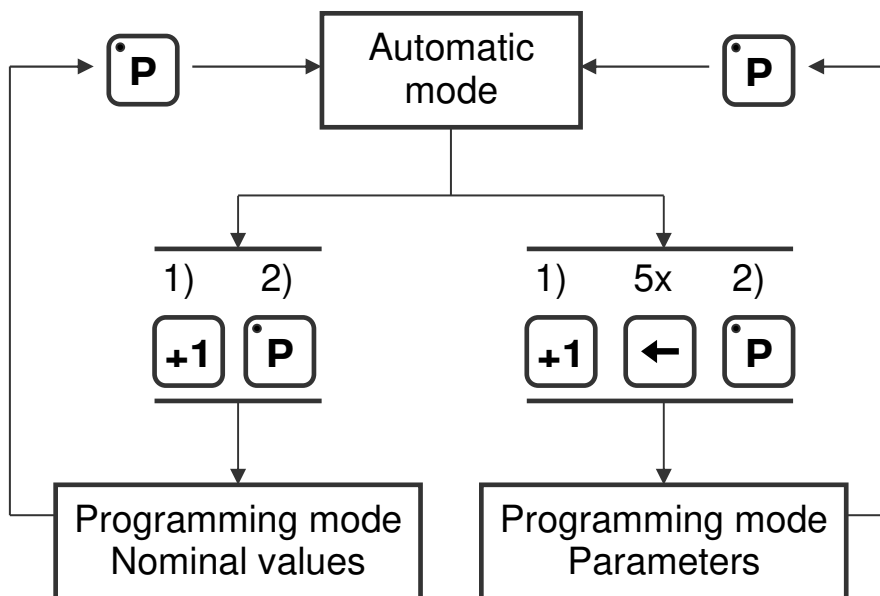
Error messages ASCII

- er 1 = parity error
- er 2 = frame error
- er 3 = overflow error
- er 4 = buffer overrun
- er 5 = number invalid
- er 6 = data invalid (outside min/max value)
- er 7 = programming mode parameters/nominal values active

3 Programming

There are three different modes of operation:

- Automatic mode
- Programming mode for nominal values
- Programming mode for parameters



- 1) hold
- 2) press once

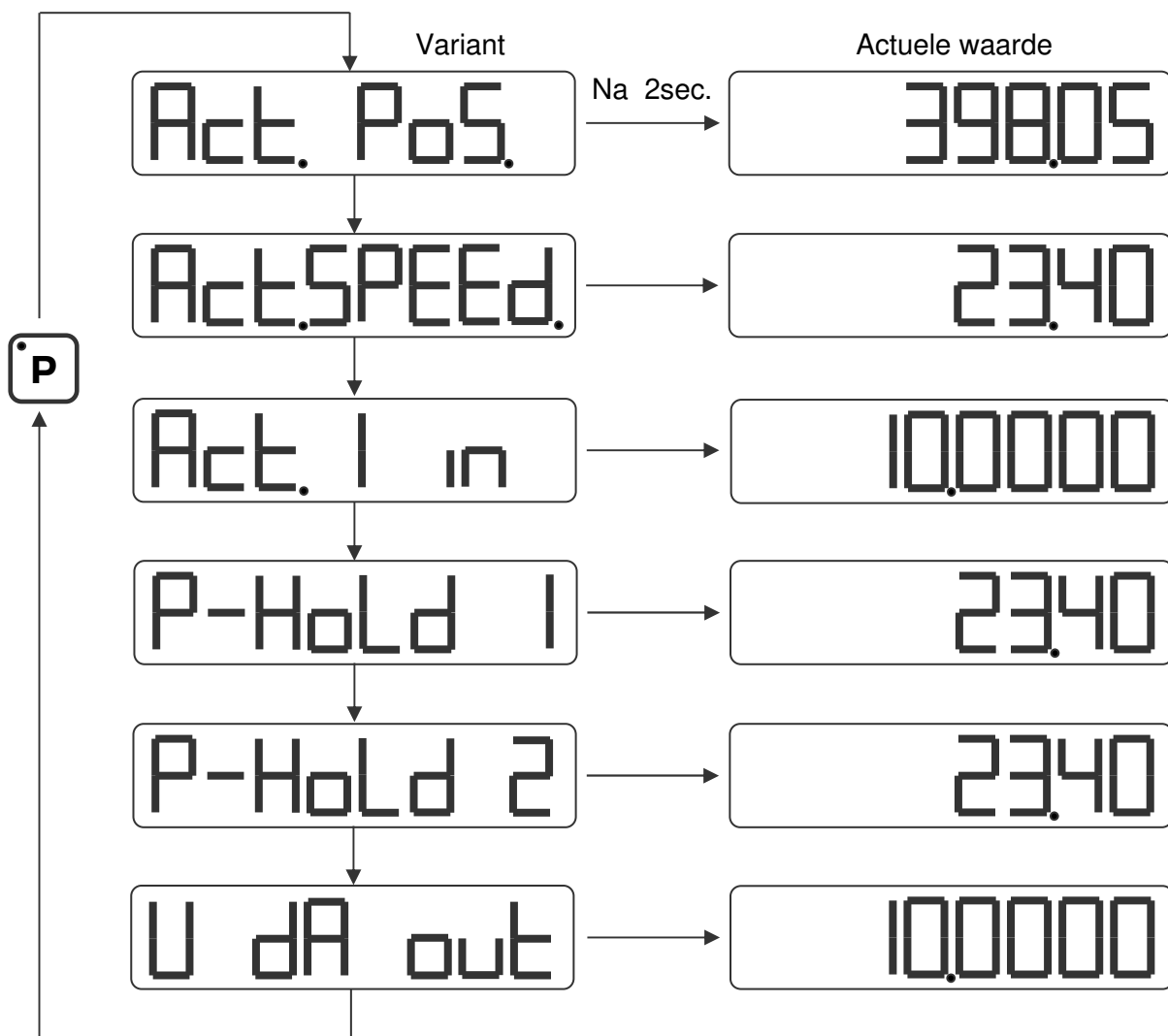
3.1 Automatic mode

In the automatic mode, depending on the type of sensor, the increments are counted or the absolute position is read and the result is shown on the display as the actual value. The velocity is calculated and can be visualized as well.

3.1.1 Monitor function

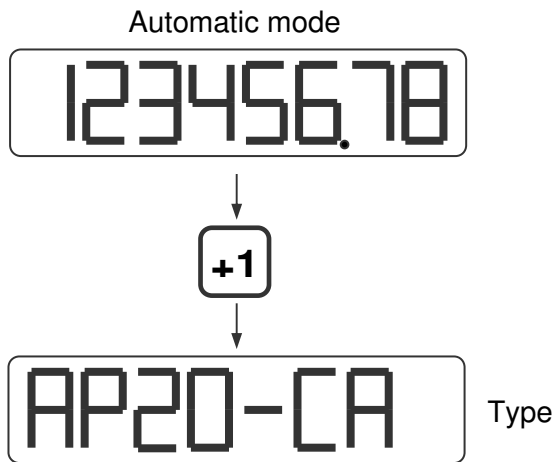
In automatic mode different variables can be displayed. By using the [P] key one can cycle through the different pages:

- Actual position
- Actual velocity
- Actual linearization
- Actual PeakHold 1
- Actual PeakHold 2
- Voltage or current of the AD (analog in)

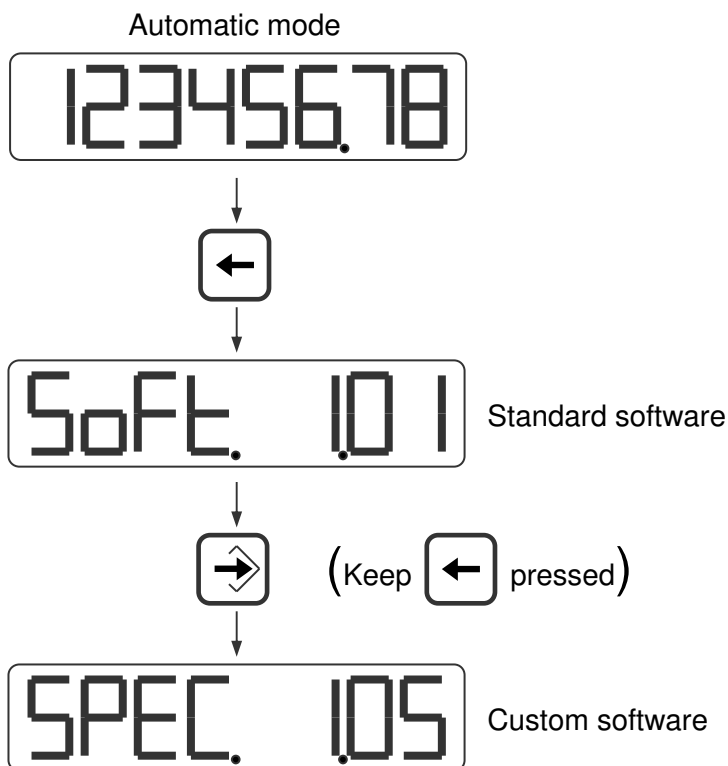


Parameter P[208] determines which option is visible after start-up.

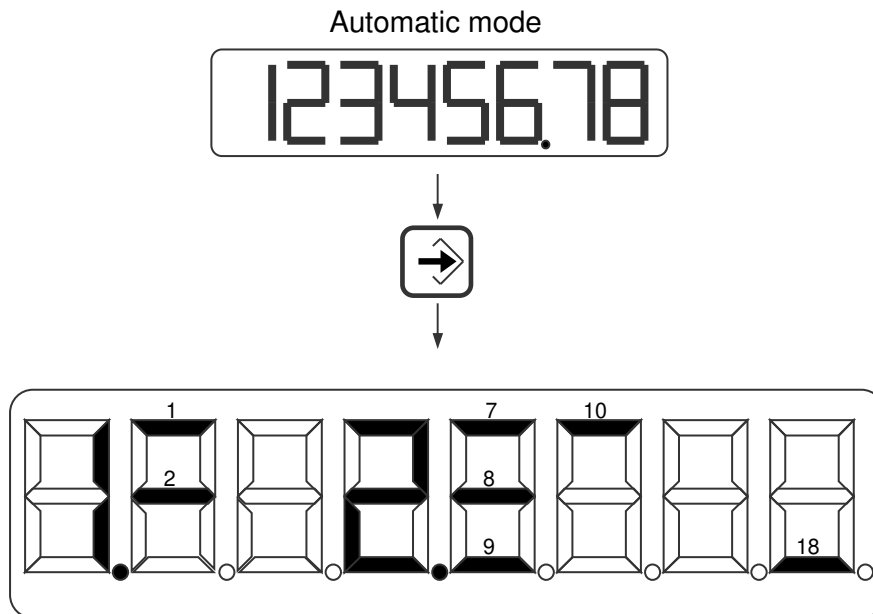
3.1.2 *Displaying the type number*



3.1.3 *Displaying the software-version*



3.1.4 Status in- and outputs



Inputs

1 = input-1
2 = input-2

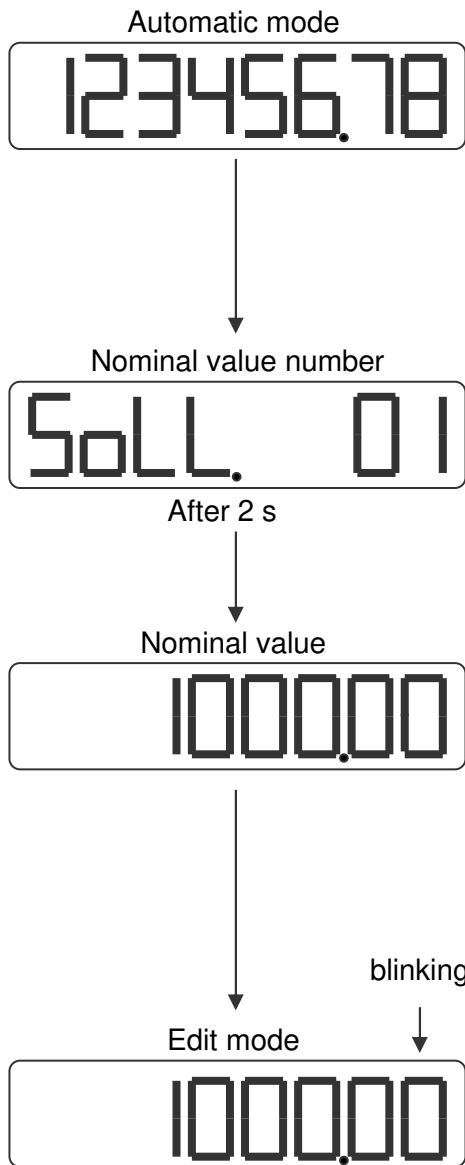
Outputs

7 = output-1
8 = output -2
9 = output -3
10 = output -4

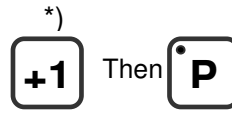
Outputs

18 = Cams active

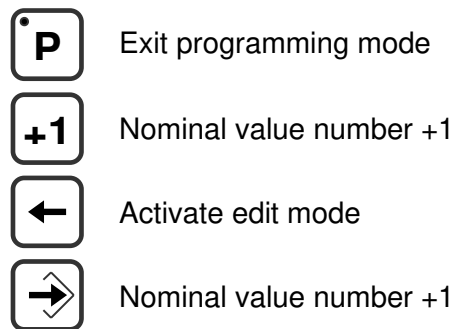
3.2 Changing nominal values



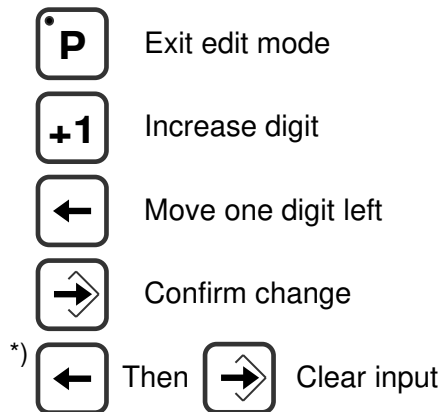
Access to nominal values



Select nominal values



Changing values



*) keep pressed

3.3 Changing parameters

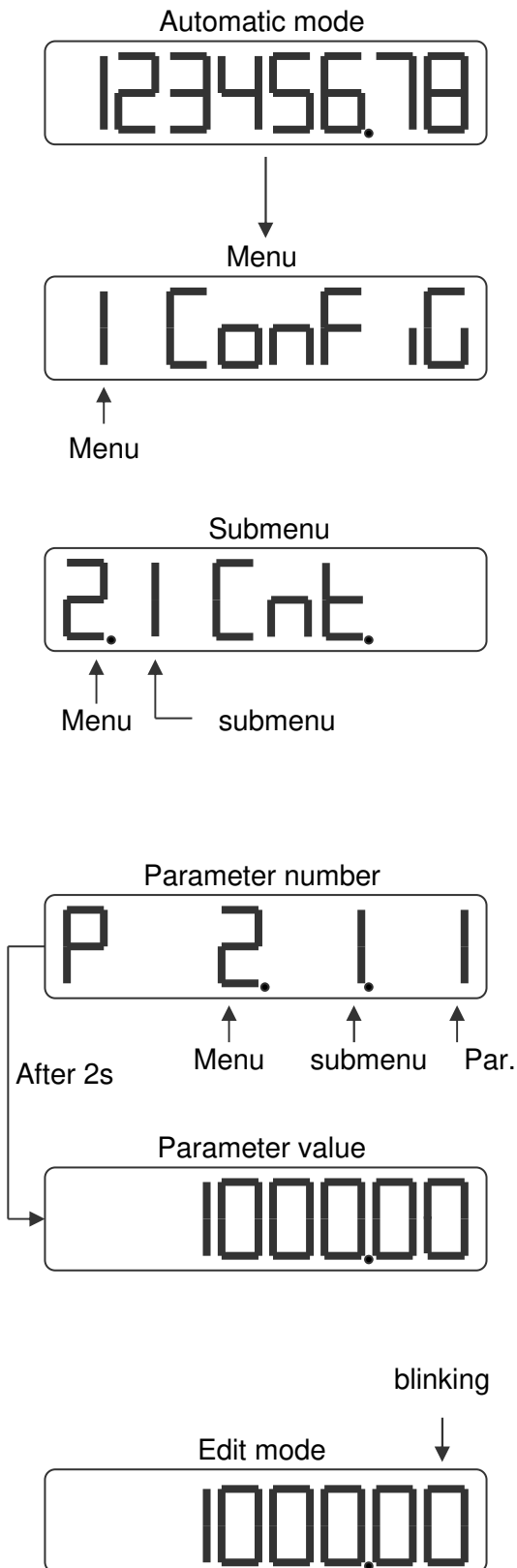
3.3.1 Menus

The parameters are displayed in different menus and submenus:

- 1 ConFiG
- 2 ActuAL
- 3 SEriAL
- 4 InPut
- 5 OutPut
 - 5.1 OP1
 -
 -
 - 5.4 OP4
- 6 Cam
 - 6.1 CA1
 -
 -
 - 6.12 CA12
- 7 An out (niet actief voor AP22)
 - 7.1 ConFG.
 - 7.2 dA-U
 - 7.3 dA-I
- 8 Lin
 - 8.1 ConFG.
 - 8.2 P 1
 -
 -
 - 8.31 P30
- 9 P-HoLd
 - 9.1 Hold1
 - 9.2 Hold2

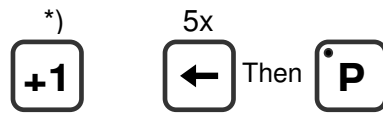
*Example:
PAr. 6.1.1 is
menu 6, submenu 1, parameter 1*

3.3.2 Input parameters

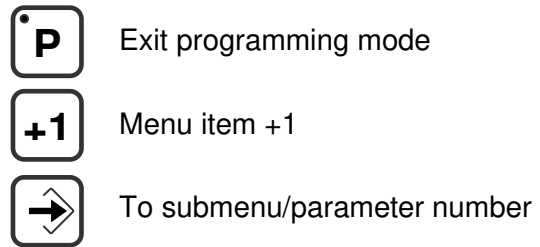


*) keep pressed down

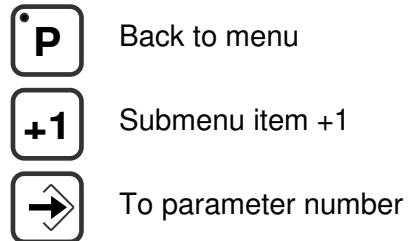
Access parameters



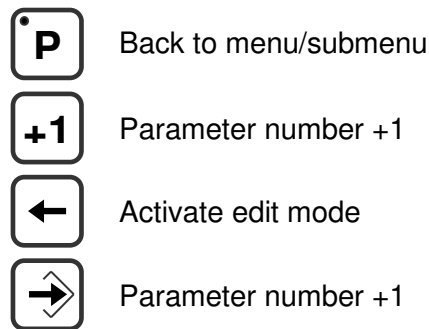
Menu selection



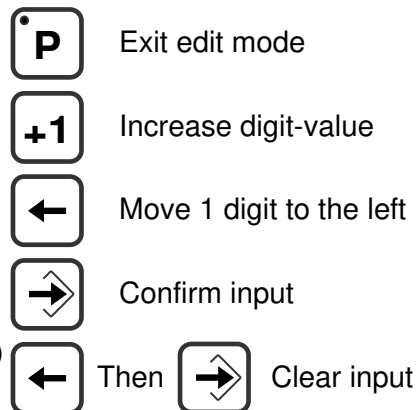
Submenu selection



Selecting parameters



Changing parameters



4 Functions

4.1 Analog input

The AP22 has an optional, galvanic isolated analog input. Using parameter P[226] it is possible to choose between a current input or a voltage input. When using the current input DIP-switch must be in the “on” position.

The analog input value (voltage or current) is converted to AWE units.
(for example 0 – 10V => 0 – 1200,0 mm)

There are 2 Parameters who configure the dynamic response of the analog input:

P[104] = measuring time

The smaller the measurement time the more dynamic the analog input will be.

P[225] = integrator

The number of cycles are programmed. The average analog value (in AWE) of the programmed cycles is calculated.

Example:

Measurement time = 60ms, integrator = 10.

The actual analog value will be refreshed every 60 ms and is the average velocity during the last 10 measurements.

4.1.1 Voltage input

The voltage input has a resolution of $305 \mu\text{V}$ and is programmable through the parameters below.

P[105] = U1 in V (input in 0,0001V units)

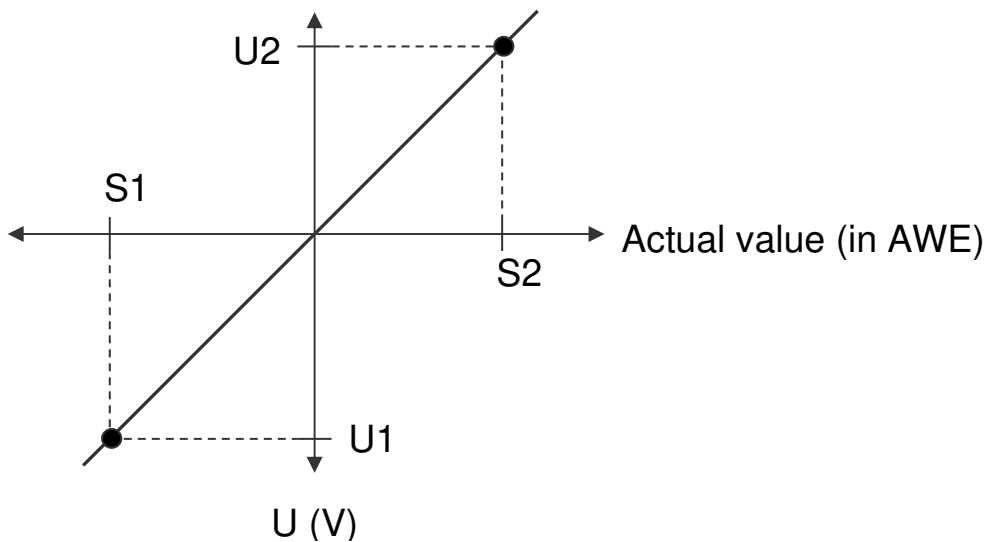
P[106] = U2 in V (input in 0,0001V units)

P[107] = S1 in AWE (actual value at U1)

P[108] = S2 in AWE (actual value at U2)

P[104] = measuring time analog

P[225] = integrator analog



4.1.2 Current input

The current input has a resolution of 610 μA and is programmable through the parameters below.

P[109] = I1 [mA] (input in 0,0001mA units)

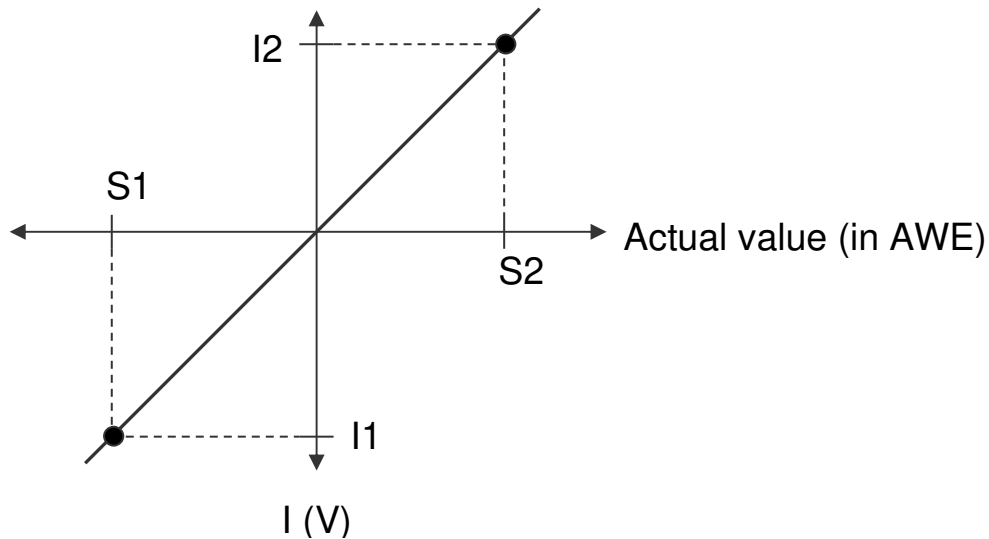
P[110] = I2 [mA] (input in 0,0001mA units)

P[111] = S1 in AWE (actual value at I1)

P[112] = S2 in AWE (actual value at I2)

P[104] = measuring time analog

P[225] = integrator analog



4.2 Velocity measurement

The velocity measurement is always active and delivers the actual speed in AWE/s.

Two parameters are necessary to configure the measurement:

P[088] = measuring time [AWE/s]

The smaller the measurement time the more dynamic the velocity measurement will be. This time is also the refreshment-interval on the display.

P[202] = integrator

The number of cycles are programmed. The average velocity (unit =AWE/s) of the programmed cycles is calculated.

Example:

Measurement time = 50ms, integrator = 10.

The actual velocity will be refreshed every 50 ms and is the average velocity during the last 10 measurements.

4.3 Justage

The justage of absolute sensors can be done in 4 ways:

- Using the parameter "Adjustment absolute value" P[005]
- Using the input input 1 or 2)
- Using parameter service functions P[207]
- Using front keys

4.3.1 Preset using the parameter "Adjustment absolute value"

This function is applicable for absolute sensors and is active when parameter "Adjustment" P[219] has been set to: **"3 PAR"**

Parameter "Adjustment absolute value" P[005] can be set to a value to adjust the encoder. This value (which can be negative and positive) will be added to the actual value and for SSI sensors compensated with the display counting range.

4.3.2 Using the input (input 1 or 2)

In this function the actual position can be preset to a certain value. The function is active when parameter "adjustment" P[219] is set to one of the following options:

- | | | |
|------------|--------------------------|----------------------|
| option: "1 | L→H Input RAM" | (rising edge input) |
| option: "2 | H→L Input RAM" | (falling edge input) |
| option: "4 | L→H Input EEPROM" | (rising edge input) |

Parameter "adjustment value" P[005] can be set to a value to which the actual position will be adjusted when input 1 or 2 detects a rising edge, possibly combined with a reference coarse signal ([P213]).

If through parameter [P213] reference coarse has been activated ([P213] <> 0)), the option **"1 - Reference coarse"** should be selected for one of the inputs 1...2 (P[249]...P[250]).

The calculated offset will be stored in either RAM or EEPROM depending on the option chosen in parameter "adjustment" P[219]. When storing into RAM-memory (option 1 and 2) this will occur on a interrupt basis and can be performed during movement. The memory however is volatile and the value will not be permanently stored.

Storing the value into EEPROM (option 4) is not initiated by an interrupt. It is recommendable to perform this only when not moving or moving very slow. This value will be stored permanently.

4.3.3 Using service parameter P[207]

When P[207] has the value 111 then the actual position is adjusted to the value of P[005] when the programming mode is left. After executing this function parameter P[207] will have the value 0 again.

4.3.4 Justage using front keys

The function preset the absolute position can be executed with the key combination [Enter] + [P].

Holding down the [Enter] key and push the [P] key once. Then the actual position is preset to the value of P[005].

This function is active when: P[219] = variant "**5 front keys**"

4.4 *ASCII protocol*

The serial port of the AP22 is able to work with an ASCII protocol.

Using the ASCII protocol, actual values can be read, parameters and nominal values can be stored and read, the status of the digital inputs and outputs can be monitored etc.

4.4.1 *Overview functions*

sc	Select AP22
r0	Read actual position (AWE)
r1	Read actual velocity (AWE/s)
r4	Read actual voltage analog input (0,1mV eenheden)
r5	Read actual current analog input (0,1 mA eenheden)
r8	Lezen actuele linearization (AWE)
ra	Read actual PeakHold 1
rb	Read actual PeakHold 2
ri	Read status inputs
ru	Read status outputs
wu	Write outputs
rp	Read parameter
wp	Write parameter (Only EEPROM)
rs	Read nominal value
ws	Write nominal value (RAM + EEPROM)
rx	Read software version
rt	Read type number
rh	Read hardware version
rf	Read error number
wf	Reset SSI error
rn	Read status bits
bp	Load and activate

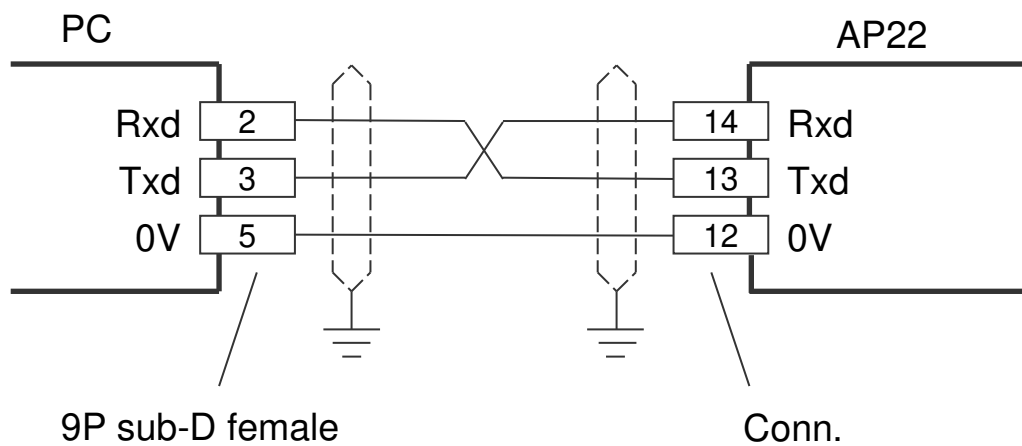
4.4.2 General

Through the ASCII protocol it is possible to communicate with the AP22.

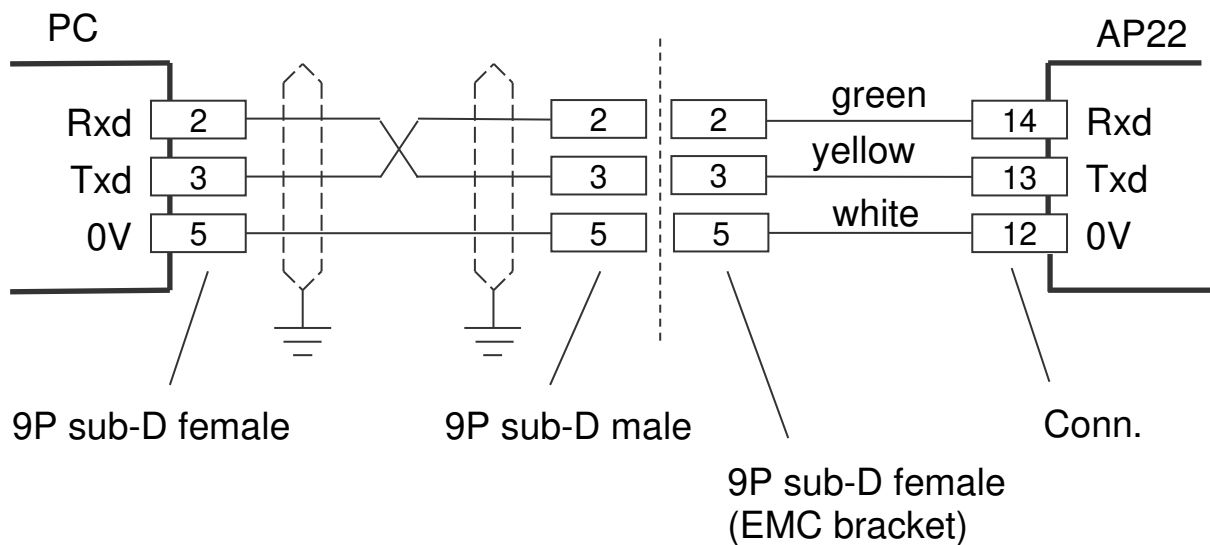
Send: Data from PC, PLC → AP22

Receive: Data from AP22 → PC, PLC

Connection without EMC bracket



Connection with EMC bracket (EMC-B02)



Send structure:

Functioncode (space) [argument 1](space)[argument 2] <CR>

Receive structure:

Functioncode (space) [argument1](space)[argument 2] <CR> <LF>

Functioncode (space) [argument1] [argument 2].

Argument 1 and 2 are depending on the function and are separated by a space.

Example:

***wp 20 250** (write value 250 to parameter 20)*

4.4.3 Functions

sc Select AP22

send: **sc xx**
receive: **sc xx**
transmitting parameter: Unit Id number

The AP22 with the unit Id number is selected, all consecutive commands are relevant for this unit.

An AP22 with unit Id number 1 will always respond.

r0 Read actual position (AWE)

send: **r0**
receive: **r0 xxxxxxxx**
transmitting parameter: none

r1 Read actual velocity (AWE/s)

send: **r1**
receive: **r1 xxxxxxxx**
transmitting parameter: none

r4 Read actual voltage analog input (0,1mV units)

send: **r4**
receive: **r4 xxxxxxxx**
transmitting parameter: none

r5 Read actual current analog input (0,1 mA units)

send: **r5**
receive: **r5 xxxxxxxx**
transmitting parameter: none

r8 Read actual linearization (AWE)

send: **r8**
receive: **r8 xxxxxxxx**
transmitting parameter: none

ra Read actual PeakHold 1 in (AWE)

send: **ra**
receive: **ra xxxxxxxx**
transmitting parameter: none

rb Read actual PeakHold 2 in (AWE)

send: **rb**
receive: **rb xxxxxxxx**
transmitting parameter: none

ri Read status inputs

send: **ri**
receive: **ri xxx**
transmitting parameter: none

B0 = input-1
B1 = input-2
B2 = 0
B3 = 0
B4 = 0
B5 = 0
B6 = 0
B7 = 0

Example: ri 03 gives the following answer:

03 → 03H, 0000 0011 B

input-1 = "1"

input-2 = "1"

ru Read status outputs

send: **ru**
receive: **ru xxx**
transmitting parameter: none

B0 = output-1
B1 = output-2
B2 = output-3
B3 = output-4

wu Write outputs

(only valid for outputs with "ASCII protocol" selected)

send: **wu xxx**
receive: **wu xxx**
transmitting parameter: data for output

B0 = output-1
B1 = output-2

Example:

Output-2 should be set to "1":

00000010B = 02H = 02D

send: wu 10

receive: wu 10

rp Read parameter

send: **rp xxx**
receive: **rp xxxxxxxx**
transmitting parameter : parameter number

Example reading parameter P[004]

send: rp 4

answer: rp 4 10000

wp write parameter (only EEPROM)

send: **wp xxx xxxxxxxxx**
receive: **wp xxx xxxxxxxxx**
transmitting parameter: parameter number and parameter value

Example writing parameter P[004] with value 185000

send: wp 4 185000

answer: wp 4 185000

Parameter will be stored to EEPROM but is not yet active.

rs Read nominal value

send: **rs xx**
receive: **rs xx xxxxxxxxx**
transmitting parameter: nominal value

Example reading nominal value 22

send: rs 22

answer: rs 22 72500

ws Write nominal value (RAM + EEPROM)

send: **ws xx xxxxxxxxx**
receive: **ws xx xxxxxxxxx**
transmitting parameter: nominal value number and nominal value

Example writing nominal value 22 with value 195200

send: wp 22 195200

answer: wp 22 195200

rx Read software version

send: **rx**
receive: **rx SW Vxx.xx SSW xx.xx**
transmitting parameter: none

SW = standard software version
SSW = special software version

Example:

send: rx

answer: rx SW 4.02 SSW 1.00

rt Read type number

send: **rt**

receive: **rt AP22-D0**

transmitting parameter: none

Example:

send: rt

answer: rt AP22-D0

rh Read hardware version

send: **rh**

receive: **rh HW x RV x**

transmitting parameter: none

rf Read error number

send: **rf**

receive: **rf xxxx**

transmitting parameter: none

When -1 returns no error is present.

Example:

send: rf

answer: rf 004 (parameter error)

or

answer: rf -1 (no error)

wf Reset SSI error

send: **wf**
receive: **wf**
transmitting parameter: none

rn Read status bits

send: **rn**
receive: **rn xxx**
transmitting parameter: none

B0 = cams active (started)
B1 = reference set

bp Load and activate

send: **bp**
receive: **bp xxx**
transmitting parameter: none

In case of an error the error number will be returned (-1 is no errors).

Example: answer: **bp -1** (no errors)
 answer: **bp 20** (error parameter 20)

4.4.4 *Error messages*

In case of an error the AP22 will sent an error message followed by an error number.

overview error messages

er 1 = parity error

er 2 = frame error

er 3 = overflow error

er 4 = buffer overrun

er 5 = number invalid

er 6 = data invalid (for example outside min/max range)

er 7 = programming mode parameters/nominal values still active

er 8 = function impossible

4.5 Cams

4.5.1 General

The AP22 has a maximum of 12 programmable cams divided over a maximum of 4 outputs. Each with 5 switching types.

Programmable functions:

- Switching type
 1. Cam with start- and end-value
 2. Greater than or equal to limit value
 3. Smaller than or equal to limit value
 4. Greater than or equal to limit value (Timer)
 5. Smaller than or equal to limit value (Timer)
- Source (actual position or actual velocity)
- Nominal value location number for start-, end-, or limit value
- Hysteresis
- Output for cam
- Switching time (only for timer cam Type 4 en 5)

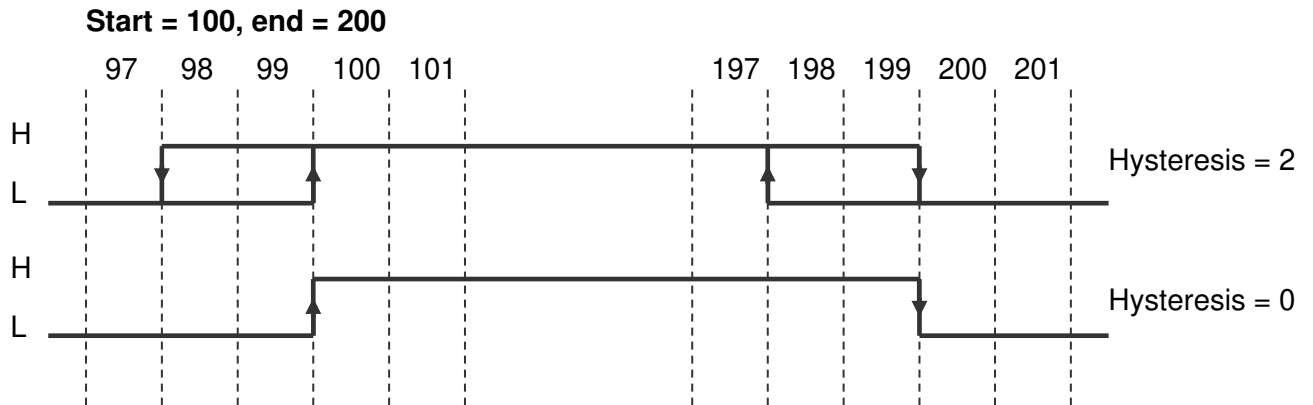
For switching type 1 there is a start- and end value. For Switching type 2...5 there is only a limit value needed.

When selecting switching type 4 and 5 the cam length is depending on the programmed time value in parameter P[120]...P[131].

Per cam one can choose whether to program the values directly into the parameters or to use a nominal value location number where the values are programmed.

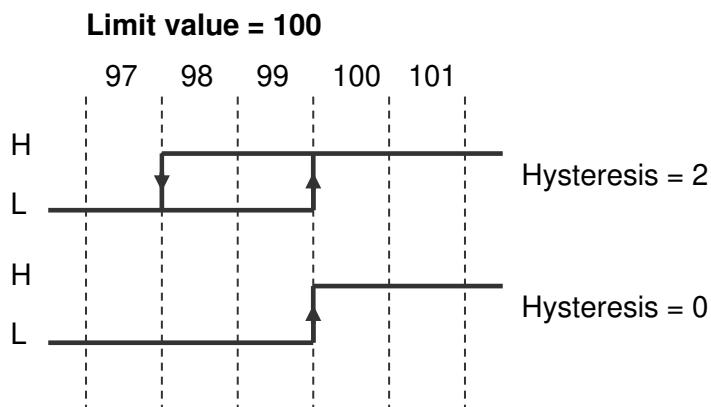
4.5.2 Switching type (1) Cam with start- and end-value

Two values are programmed, a start value and an end-value.



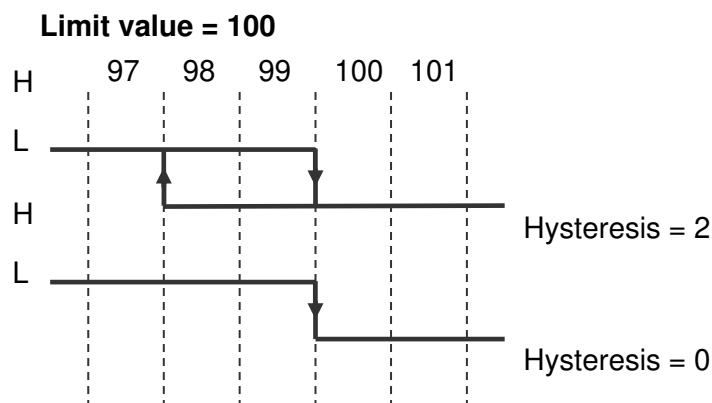
4.5.3 Switching type (2) Greater than or equal to limit value

Only one limit value needs to be programmed.



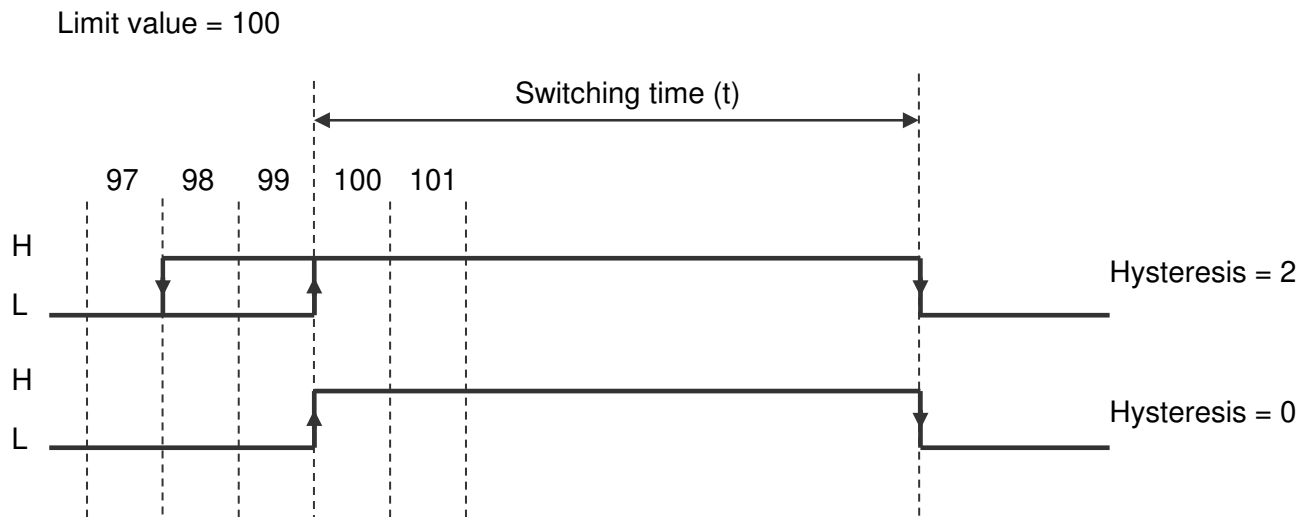
4.5.4 Switching type (3) Smaller than or equal to limit value

Only one limit value needs to be programmed.



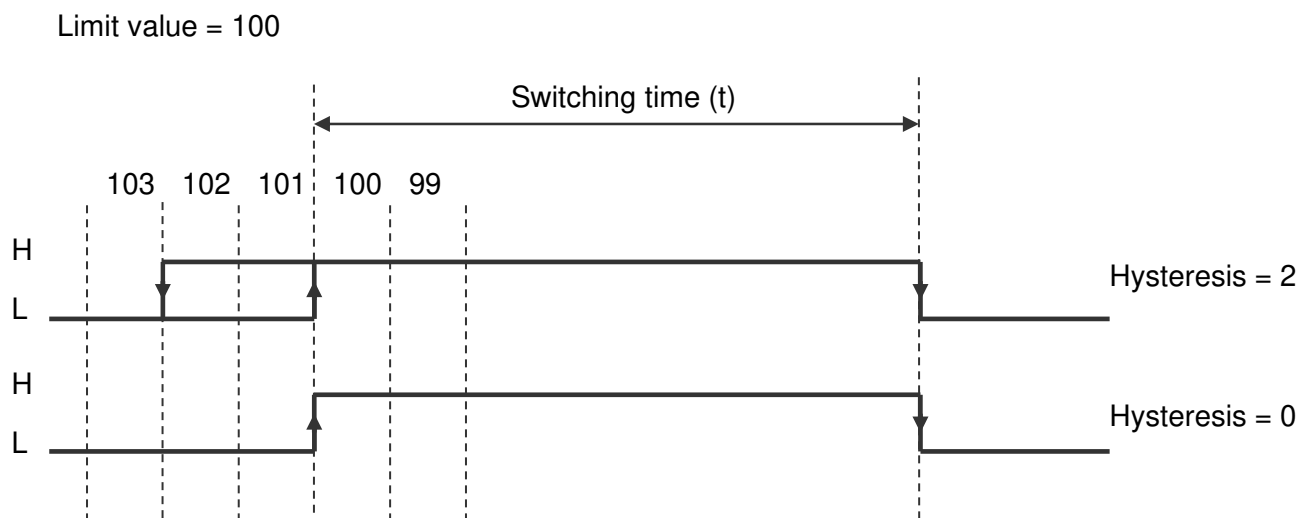
4.5.5 Switching type (4) Greater than or equal to limit value (Timer)

Limit value and switching time need to be programmed. This cam is activated when the actual value is greater than or equal to the limit value (counting upwards) and is a retriggerable timer.



4.5.6 Switching type (5) Smaller than or equal to limit value (Timer)

Limit value and switching time need to be programmed. This cam is activated when the actual value is smaller than or equal to the limit value (counting downwards) and is a retriggerable timer.



4.5.7 Dynamic cams

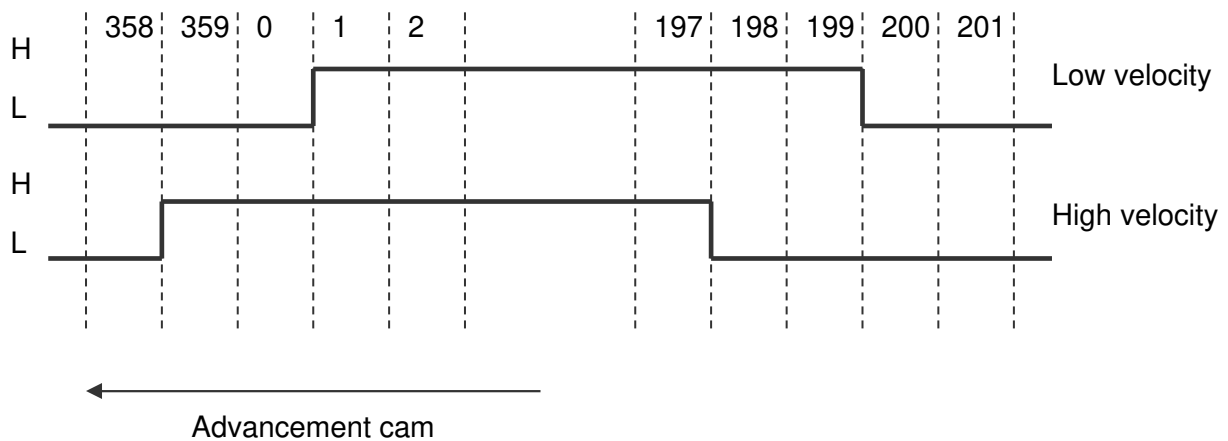
To compensate for actions with a static time, for example the switch time of a valve, it is possible to program a time for each output individually. The cams will be shifted according to this programmed time.

This function is only available for cams with (P[256] ... P[267] = 1, 4 or 5) and when the source of this cam is the actual position (P[280] ... P[291] = 1).

When working with a counting range the possibility of cams shifting over the zero-point is taken into account.

Example:

startvalue: 1
endvalue: 200
hysteresis: 0
counting range display: 360



4.5.8 *Start/stop cam*

The outputs for the cams can be enabled or disabled. If for one of the inputs 1..6 the function start/stop cams has been chosen.

Start/stop cams with one signal

Input-x = option "**5 start/stop cams**" (high = cams enabled)

Start/stop cams with double signal

Input-x = option "**6 start cams**" (rising edge = enable cams)

Input-x = option "**7 stop cams**" (rising edge = disable cams)

4.5.9 *Output "Cams active"*

On one of the outputs 1...4 the signal "cams active" can be generated by choosing option "**4 cams active**" (high = cams enabled).

4.6 Linearization

The linearization function allows to display and process nonlinear motions. The actual display position (sensor) position or actual velocity is converted into an additional value "Actual linearization". This additional value can serve as a source for the cams function and analog output.

Parameter "Source" P[246]

optie "0 Inactive"

optie "1 Actual position"

optie "2 Actual speed"

The linearization function works with a tabel of max. 30 points (P1...P30). Every point has a X (Pn-X) and Y (Pn-Y) value. The X value represents the actual (sensor) position or velocity and the Y value represents the desired actual (sensor) position or velocity called "Actual linearization". Interpolation takes place between these points

Parameter P[247] determines the number of active points and can be programmed with a value of 2 ...30 points. The value of the actual linearization can be displayed in the monitor function. P[248] determines the number of decimals shown.

For linearization 3 modes are available.

Parameter "mode" P[241]

optie "0 4-Quadrant"

optie "1 Mirror Y-axis"

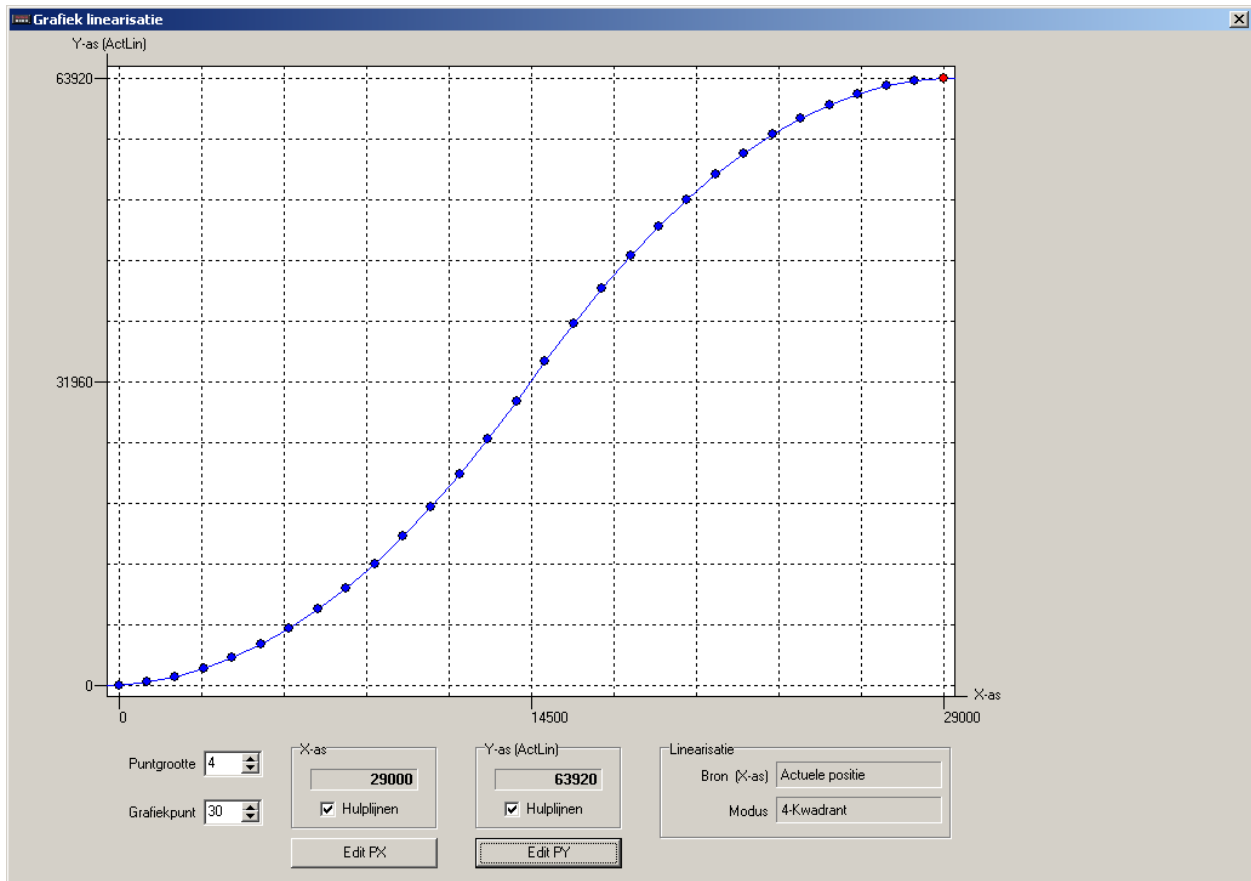
optie "2 Mirror XY-axis"

4.6.1 mode 0 "4-Quadrant"

This is the standard mode which can build every possible curve. Both the X-axis and the Y-axis may contain negative values.

Condition:

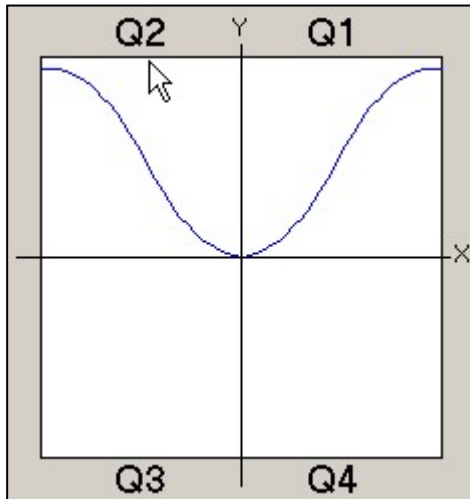
$P1-X < P2-X < P3X < \dots < P30-X$



Screenshot from PC programm DST2

4.6.2 mode 1 “Mirror Y-axis”

In this mode, the entered curve is mirrored and copied over the Y-axis, so that this curve is the same for the negative X-axis values. Q1 is the entered curve.



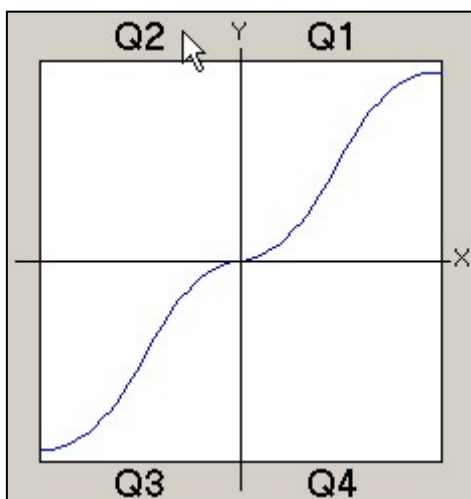
Conditions:

$$P1-X < P2-X < P3X < \dots < P30-X)$$

$$P1-X = 0$$

4.6.3 mode 2 “Mirror XY-axis”

In this mode, the entered curve is mirrored and copied over the X-axis and Y-axis, so that this curve is the same for the negative X-axis and Y-axis values. Q1 is the entered curve.



Conditions:

$$P1-X < P2-X < P3X < \dots < P30-X)$$

$$P1-X = 0$$

$$P1-Y = 0$$

4.7 PeakHold function

4.7.1 general

The PeakHold function can hold the minimum or maximum value of the selected source like actual- position, velocity or linearization.

On power-up or external reset the PeakHold value will be set to the actual value of the selected source.

The PeakHold value can also serve as a source for the cams function.

4.7.2 PeakHold 1

Parameter "Function PeakHold 1" P[242]

option "0 no function"

option "1 detect Peak high"

option "2 detect Peak low"

Parameter "Source PeakHold 1" P[243]

option "0 actual position"

option "1 actual velocity"

option "2 actual lineariaation"

4.7.3 PeakHold 2

Parameter "Function PeakHold 2" P[244]

option "0 no function"

option "1 detect Peak high"

option "2 detect Peak low"

Parameter "Source PeakHold 2" P[245]

option "0 actual position"

option "1 actual velocity"

option "2 actual linearizsation"

5 Parameters

General lay-out:

PAR.	PAR Nr:	Possible values (bold is the standard value)
Basic description		
Description of possible values		

5.1 Menu 1 Config

PAR: 1.0.1	P[088]	0 ... 40 ... 2500
Measuring time velocity [AWE/s] (equal to refreshment time display)		
X.XXX (sec) input 0 .. 1.000s		

PAR: 1.0.2	P[202]	0 ... 10 ... 20
Integrator velocity		
Actual velocity is the average from the number of measurements		
0 = not active		
1...20 number of measurements		

PAR: 1.0.3	P[203]	0 ... 6
Number of decimals		
0 = none		
1 = X.X		
2 = X.XX		
3 = X.XXX		
4 = X.XXXX		
5 = X.XXXXX		
6 = X.XXXXXX		

PAR: 1.0.4	P[204]	0 ... 1
Store function		
0 = no function		
1 = display		

PAR: 1.0.5	P[205]	0 ... 2
Store signal		
0 = high active		
1 = low active		

PAR: 1.0.6	P[206]	0 ... 1
Power failure protection (only AP20)		
0 = not active		
1 = active		

PAR: 1.0.7	P[207]	0 ... 123
Service functions		
Only possible to activate through keyboard AP22		
Value is automatically set to 0 after executing the service function		
0 = not active		
123 = set default parameters		
111 = adjusting absolute sensors or preset function counter and freq.		

PAR: 1.0.8	P[208]	0 ... 4
Default monitor function		
Determines the default which is visible after start-up.		
0 = Actual position		
1 = Actual velocity		
2 = Actual linearization		
3 = actual PeakHold 1		
4 = actual PeakHold 2		

PAR: 1.0.9	P[201]	0 ... 3
Display multiplier		
For display value actual position + actual velocity		
0 = no function		
1 = actual position + actual velocity x 0,1		
2 = actual position + actual velocity x 0,01		
3 = actual position + actual velocity x 0,001		

5.2 Menu 2 Actual

PAR: 2.3.1	P[005]	-9999999 ... 0 ... 99999999
Adjustment absolute value		
XXXXXXXX		

PAR: 2.3.2	P[219]	0 ... 5
Adjustment		
0 = no function		
1 = rising edge (input 1 or 2) only temporary in RAM (on interrupt)		
2 = falling edge (input 1 or 2) only temporary in RAM (on interrupt)		
3 = set with parameter adjustment value		
4 = rising edge (input 1 or 2) permanent in EEPROM (not on interrupt)		
5 = front keys		

PAR: 2.3.3	P[213]	0 ... 2
Reference coarse		
0 = no function		
1 = high signal		
2 = low signal		

PAR: 2.3.4	P[227]	0 ... 1
Min/max function		
0 = no function		
1 = min/max		

PAR: 2.3.5	P[113]	-9999999 ... 0 ... 99999999
Min. Value		
-XXXXXXXX (AWE)		

PAR: 2.3.6	P[114]	-9999999 ... 0 ... 99999999
Max. value		
-XXXXXXXX (AWE)		

PAR: 2.3.7	P[226]	0 ... 1
Selection AD input		
0 = voltage		
1 = current		

PAR: 2.3.8	P[104]	0 ... 10 ... 100
Measuring time analog input		
X.XX (sec) input 0 is 1.00s		

PAR: 2.3.9	P[225]	0 ... 5 ... 50
Integrator analog input		
Actual value is the average from the number of measurements		
0...50 (number of measurements) input 0 = 1		

PAR: 2.3.10	P[105]	-100000 ... 99999
U1 Voltage		
-XX.XXXX (V)		

PAR: 2.3.11	P[106]	-99999 ... 100000
U2 Voltage		
-XX.XXXX (V)		

PAR: 2.2.12	P[107]	-9999999... -100000 ... 99999999
S1 Voltage		
-XXXXXXXX (AWE)		

PAR: 2.3.13	P[108]	-9999999 ... 100000 ... 99999999
S2 Voltage		
-XXXXXXXX (AWE)		

PAR: 2.3.14	P[109]	-200000 ... 199999
I1 Current		
-XX.XXXX (mA)		

PAR: 2.3.15	P[110]	-199999 ... 200000
I2 Current		
-XX.XXXX (mA)		

PAR: 2.3.16	P[111]	-9999999... -200000 ... 99999999
S1 Current		
-XXXXXXXX (AWE)		

PAR: 1.3.17	P[112]	-9999999 ... 200000 ... 99999999
S2 Current		
-XXXXXXXX (AWE)		

5.3 Menu 3 Serial

PAR: 3.0.1	P[236]	0 ... 31
Unit adress		
XX		

PAR: 3.0.2	P[237]	0 ... 1 ... 4
Baudrate		
0 = 9600		
1 = 19200		
2 = 28800		
3 = 38400		
4 = 57600		

PAR: 3.0.3	P[238]	0 ... 1
Number of stopbits		
0 = 1 stopbit		
1 = 2 stopbits		

PAR: 3.0.4	P[239]	0 ... 2
Parity		
0 = none		
1 = odd		
2 = even		

PAR: 3.0.5	P[240]	0 ... 1
Protocol		
0 = no function		
1 = ASCII		

5.4 Menu 4 Input

INPUT-1

PAR: 4.0.1	P[249]	0 ... 13
Function input-1		
0 = no function		
1 = coarse		
2 = reference fine		
3 = store		
4 = error reset SSI		
5 = start/stop cams		
6 = start cams		
7 = stop cams		
8 = lock input nominal values		
9 = lock input parameters		
10 = lock input nominal values + parameters		
11 = DAC Set Smin		
12 = DAC Set Smax		
13 = reset PeakHold 1/2		

INPUT-2

PAR: 4.0.2	P[250]	0 ... 13
Function input-2		
XX (see input-1)		

5.5 Menu 5 Output

5.5.1 Submenu 5.1 – 5.4 Op1...4

OUTPUT 1...4

PAR: 5.x.1	P[252]...P[255]	0 ... 7
Function output-1		
0 = cam		
1 = cam inverted		
2 = SSI error (high = no error) only AP21		
3 = Start/Stop error (no magnet or time-out) only AP23		
4 = reference/adjustment set		
5 = cams active		
6 = ASCII protocol		
7 = Counting direction (high = downwards counting)		

PAR: 5.x.2	P[094]...P[097]	0 ... 5000
Dynamic cams (only when with cam function 1, 4 and 5 and source = actual position)		
X.XXX (sec) input 0 = no function		

5.6 Menu 6 Cam

5.6.1 Submenu 6.1 ... 6.12 CA1...12

CAM-1...12

PAR: 6.x.1	P[256]...P[267]	0 ... 5
Cam function		
0 = no function		
1 = range		
2 = actual position \geq limit value		
3 = actual position \leq limit value		
4 = actual position \geq limit value (Timer)		
5 = actual position \leq limit value (Timer)		

PAR: 6.x.2	P[280]...P[291]	0 ... 4
Source		
0 = actual position		
1 = actual velocity		
2 = actual linearization		
3 = actual PeakHold 1		
4 = actual PeakHold 2		

PAR: 6.x.3	P[304]...P[315]	0 ... 12
Source cam begin / limit value (limit value if cam function = 2 or 3)		
0 = parameters cam begin		
1...12 = Nominal value 1...12		

PAR: 6.x.4	P[292]...P[303]	0 ... 12
Source cam end		
0 = parameters cam end		
1...12 = Nominal value 1...12		

PAR: 6.x.5	P[007]...P[018]	-9999999 ... 1000 ... 99999999
Cam begin / limit value (limit value if cam function = 2 or 3)		
-XXXXXXXX		

PAR: 6.x.6	P[031]...P[042]	-9999999 ... 2000 ... 99999999
Cam end		
-XXXXXXXX		

PAR: 6.x.7	P[055]...P[066]	0 ... 999999
Hysteresis cam		
XXXXXX		

PAR: 6.x.8	P[268]...P[279]	0 ... 4
Assign cam to output		
0 = no output		
1...4 = output 1-4		

PAR: 6.x.9	P[120]...P[131]	0 ... 5000
Cam switching time (only with cam function 4 en 5)		
X.XXX (sec) ingave 0 = 0,1 sec		

5.7 Menu 8 Linearization

5.7.1 Submenu 8.1 Config

PAR: 8.1.1	P[246]	0 ... 2
Source		
0 = inactive		
1 = actual position		
2 = actual velocity		

PAR: 8.1.2	P[247]	2 ... 10... 30
Number of points		
XX		

PAR: 8.1.3	P[248]	0 ... 6
Number of decimals for actual position linearization		
0 = geen		
1 = X.X		
2 = X.XX		
3 = X.XXX		
4 = X.XXXX		
5 = X.XXXXX		
6 = X.XXXXXX		

PAR: 8.1.4	P[241]	0 ... 2
Mode		
0 = 4-Quadrant		
1 = Mirror X-as		
2 = Mirror XY-as		

5.7.2 Submenu 11.2 ... 11.31 P1...P30

P1...30 → x = 2...31

PAR: 8.x.1	P[140]...P[169]	-9999999 ... 0 ... 99999999
P1-X		
-XXXXXXXX		

PAR: 8.x.2	P[170]...P[199]	-9999999 ... 0 ... 99999999
P1-Y		
-XXXXXXXX		

5.8 Menu 9 PeakHold

5.8.1 Submenu 9.1 PeakHold 1

PAR: 9.1.1	P[242]	0 ... 2
Function PeakHold 1		
0 = non function		
1 = detect peak high		
2 = detect peak low		

PAR: 9.1.2	P[243]	0 ... 2
Source PeakHold 1		
0 = actual position		
1 = actual velocity		
2 = actual linearization		

5.8.2 Submenu 9.2 PeakHold 2

PAR: 9.2.1	P[244]	0 ... 2
Function PeakHold 2		
0 = geen functie		
1 = detectie peak high		
2 = detectie peak low		

PAR: 9.2.2	P[245]	0 ... 2
Source PeakHold 2		
0 = actual positie		
1 = actual velocity		
2 = actual linearization		

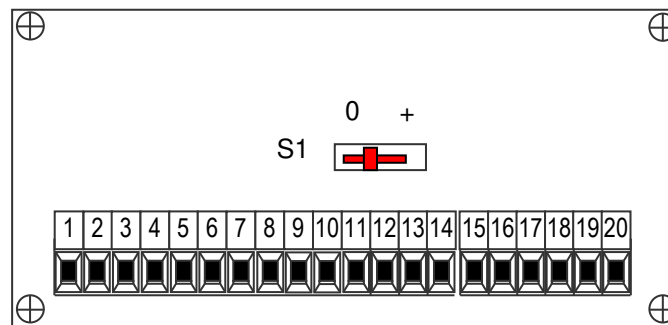
5.9 Overview parameters

<u>No</u>	<u>Description</u>	<u>Menu</u>
[000]...[004]	= no function	
[005]	= Adjustment value	2.3.1
[019]	= no function	
[007]...[018]	= Cam begin / limit value	6.1.5...6.12.5
[019]...[030]	= no function	
[031]...[042]	= Cam end	6.1.6...6.12.6
[043]...[054]	= no function	
[055]...[066]	= Hysteresis cam	6.1.7...6.12.7
[067]...[087]	= no function	
[088]	= Measuring time velocity	1.0.1
[089]...[093]	= no function	
[094]...[097]	= Dynamic cam output 1...4	5.1.2...5.4.2
[098]...[103]	= no function	
[104]	= Measuring time analog input	2.3.8
[105]	= U1 Voltage	2.3.10
[106]	= U2 Voltage	2.3.11
[107]	= S1 Voltage	2.3.12
[108]	= S2 Voltage	2.3.13
[109]	= I1 Current	2.3.14
[110]	= I2 Current	2.3.15
[111]	= S1 Current	2.3.16
[112]	= S2 Current	2.3.17
[113]	= min. value AD	2.3.5
[114]	= max. value AD	2.3.6
[115]...[119]	= no function	
[120]...[131]	= Cam switching time	6.1.9...6.12.9
[132]...[139]	= no function	
[140]...[169]	= Pn-X (linearization)	8.2.1...8.31.1
[170]...[199]	= Pn-Y (linearization)	8.2.2...8.31.2
[200]	= no function	
[201]	= Display multiplier	1.0.9
[202]	= Integrator velocity	1.0.2
[203]	= Number of decimals	1.0.3
[204]	= Store function	1.0.4
[205]	= Store signal	1.0.5
[206]	= Power failure protection	1.0.6
[207]	= Service functions	1.0.7
[208]	= Default monitor function	1.0.8
[209]	= Counting direction for justage	2.1.7
[210]	= no function	

No	Description	Menu
[211]	= Counting direction	2.1.2
[212]	= no function	
[213]	= Reference coarse	2.3.3
[214]...[218]	= no function	
[219]	= Adjustment	2.3.2
[220]...[224]	= no function	
[225]	= Integrator analog input	2.3.9
[226]	= AD function	2.3.7
[227]	= min/max function AD	2.3.4
[228]...[235]	= no function	
[236]	= Unit adress	3.0.1
[237]	= Baudrate (RS232)	3.0.2
[238]	= Stopbits (RS232)	3.0.3
[239]	= Parity (RS232)	3.0.4
[240]	= Protocol (RS232)	3.0.5
[241]	= Mode linearization	8.1.4
[242]	= Function PeakHold 1	9.1.1
[243]	= Source PeakHold 1	9.1.2
[244]	= Function PeakHold 2	9.2.1
[245]	= Source PeakHold 2	9.2.2
[246]	= Source (linearization)	8.1.1
[247]	= Number of points (linearization)	8.1.2
[248]	= Number of decimals (linearization)	8.1.3
[249], [250]	= Function input 1...2	4.0.1/4.0.2
[251]	= no function	
[252]...[255]	= Function output1...4	5.0.1...5.0.4
[256]...[267]	= Cam function	6.1.1...6.12.1
[268]...[279]	= assign cam to output	6.1.8...6.12.8
[280]...[291]	= Source cam	6.1.2...6.12.2
[292]...[393]	= Source for cam end	6.1.4...6.12.4
[304]...[315]	= Source for cam begin/limit value	6.1.3...6.12.3
[316]...[399]	= no function	

6 Connections

Connections on the rear

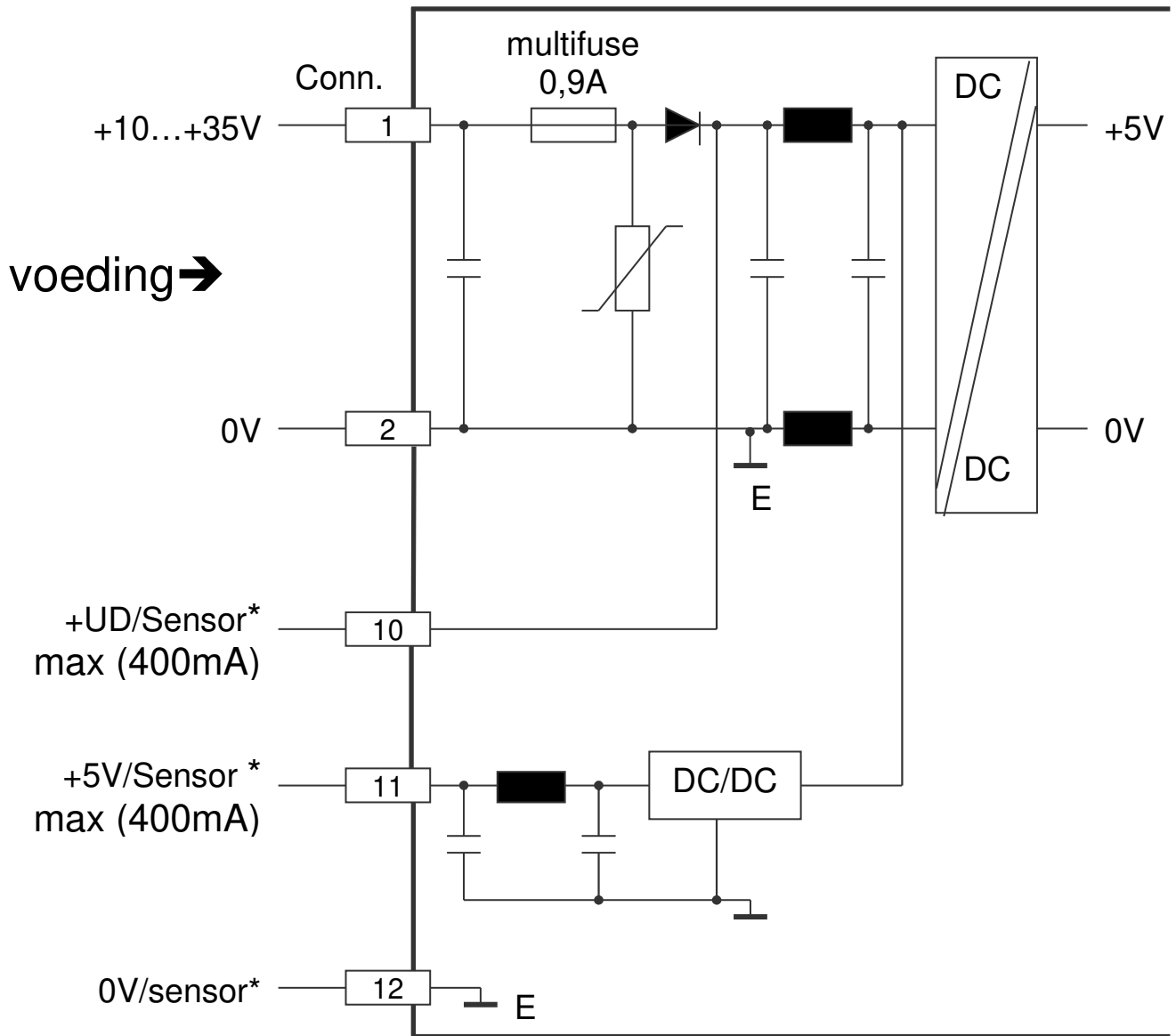


6.1 Overview clamp connections

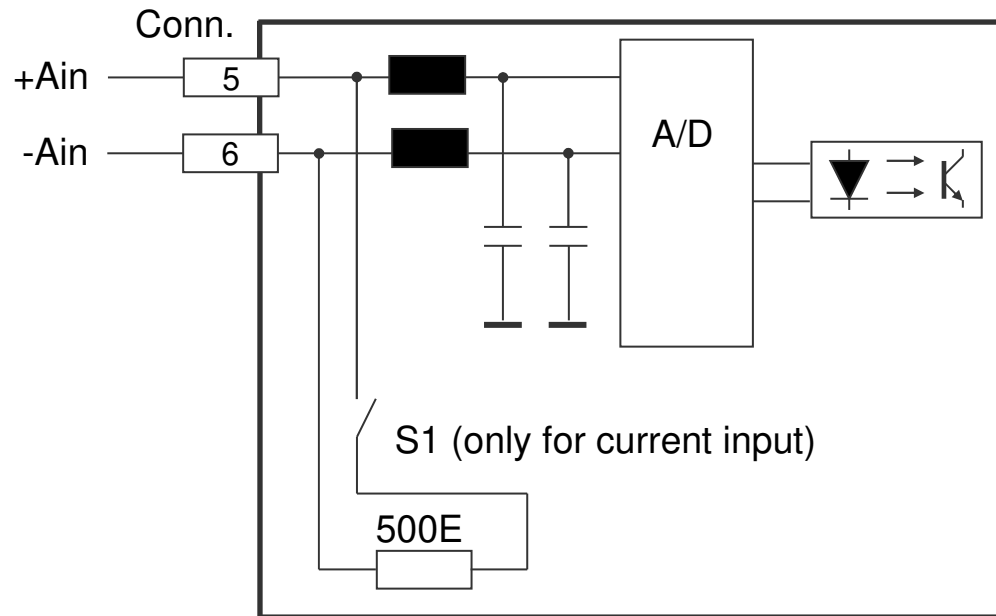
1. +10...+35V supply
2. 0V supply
3. K1 (geen functie)
4. /K1 (geen functie)
5. Analooq+
6. Analooq-
7. Input-1
8. Input-2
9. common for inputs (0V)
10. +10...35V DC supply output for sensor
11. +5V DC supply output for sensor
12. 0V for sensor
13. TxD
14. RxD

15. Output -1
16. Output -2
17. Output -3
18. Output -4
19. 0V for outputs
20. +U for outputs

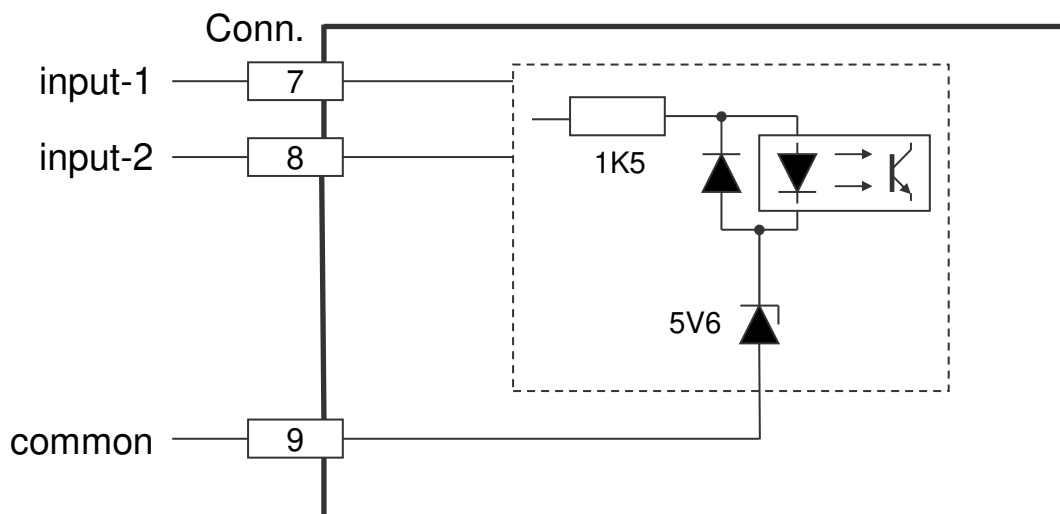
6.2 Supply



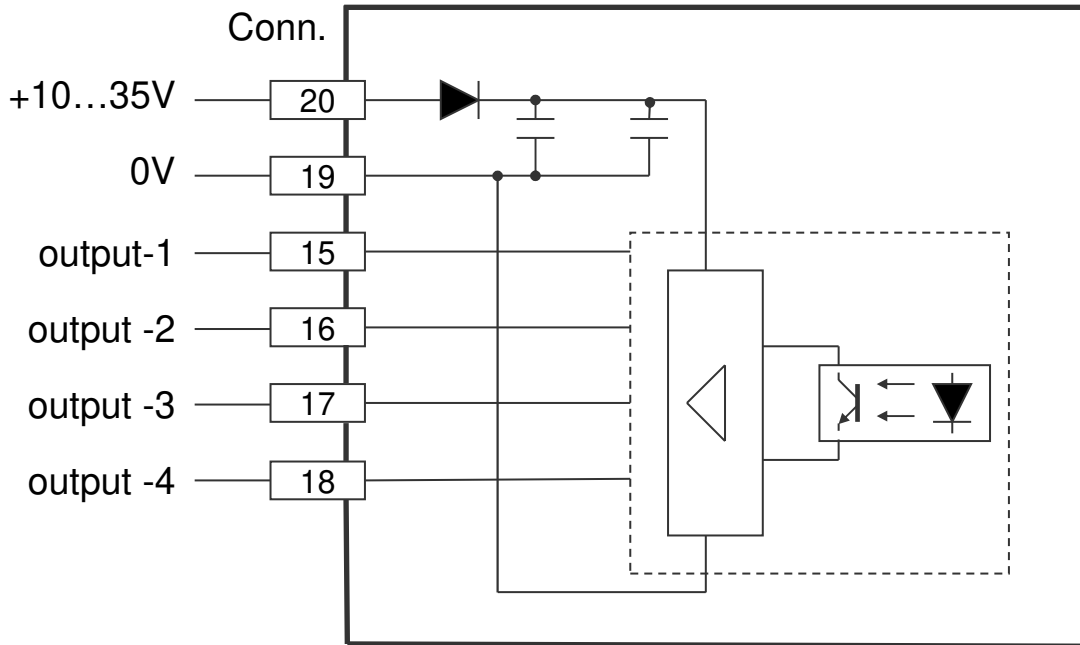
6.3 Analog input



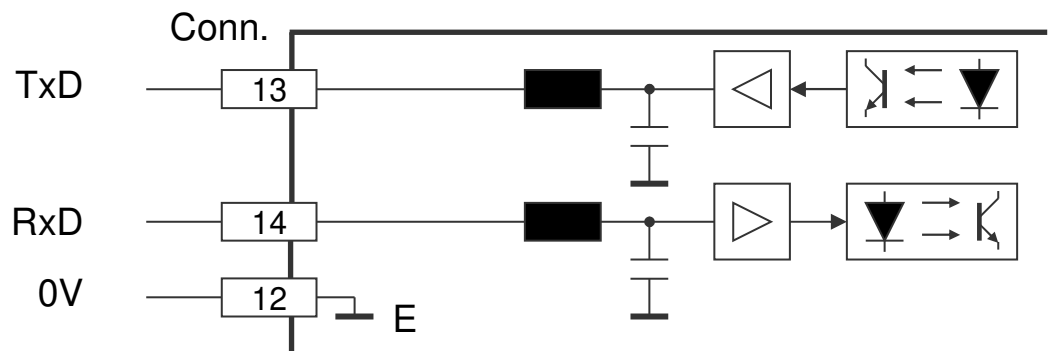
6.4 Digital inputs



6.5 Digital outputs



6.6 RS232



7 Technical specifications

7.1 Specifications

- Supply voltage	10...35V DC
current consumption	< 150mA
- Output voltage	for external encoder
+UD	max 400mA depending on supply voltage
+5V	max 400mA
- Processor	
μController	XC167
Data memory	EEPROM
Cycle time	250μS (fixed)
Counting range	-9999999...+99999999
- Digital inputs 1...2	optically isolated
	low: 0...+5V
	high: +10V...+35V
input resistor	appr. 1.8kOhm at 24V
- Digital outputs 1...4	optically isolated, N FET, short-circuit proof
I _{max}	500 mA (min load 200 μA)
Supply voltage	35V max.
- Voltage input	galvanically isolated
range	max. -10V ... +10V
resolution	305 μV
offset-temp. coeff.	< 20 ppm/ °C
R _{in}	30 Kohm
- Current input	galvanically isolated
range	max. -20mA ... +20mA
resolution	610 μA
offset-temp. coeff.	< 20 ppm/ °C
R _{in}	500 Ohm
- Serial communication	RS232 C

- Display
digit height 8 decades 7-segments LED
14 mm
- Temperature range 0...50°C
- Connection diameter 1,6 mm² (raster 3,81mm)
- Electromagnetic
compatibility in accordance with guideline
emission 2004/108/EC
immunity NEN-EN-IEC61000-6-3:2007
NEN-EN-IEC61000-6-3:2005
- Weight < 0.25 kg
- Sealing front IP50, with protective hood IP54
rear IP20

7.2 Typekey

AP22- X 0



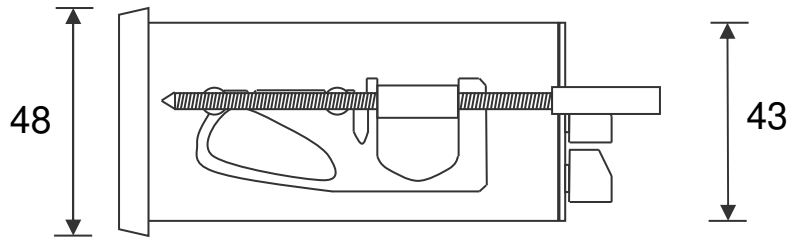
Digital outputs

0 = no digital outputs

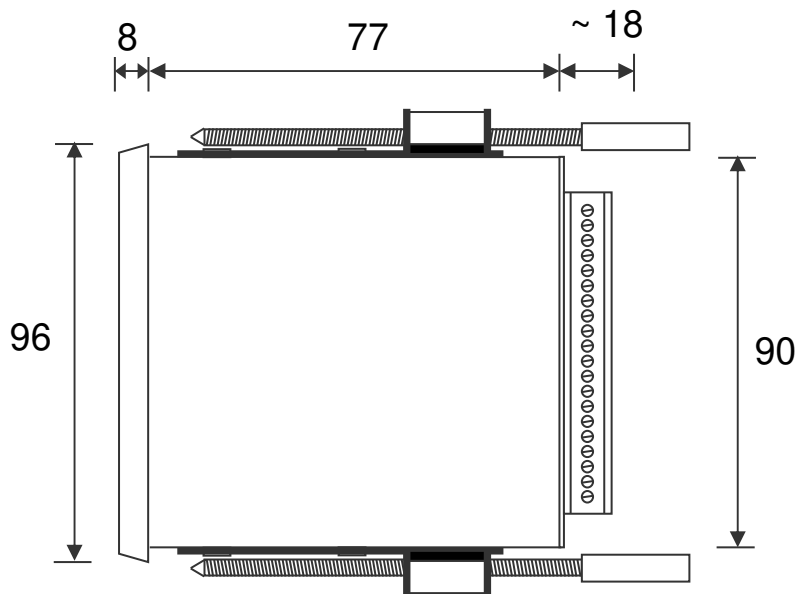
D = 4 digital outputs

7.3 Dimensions AP22

Side view

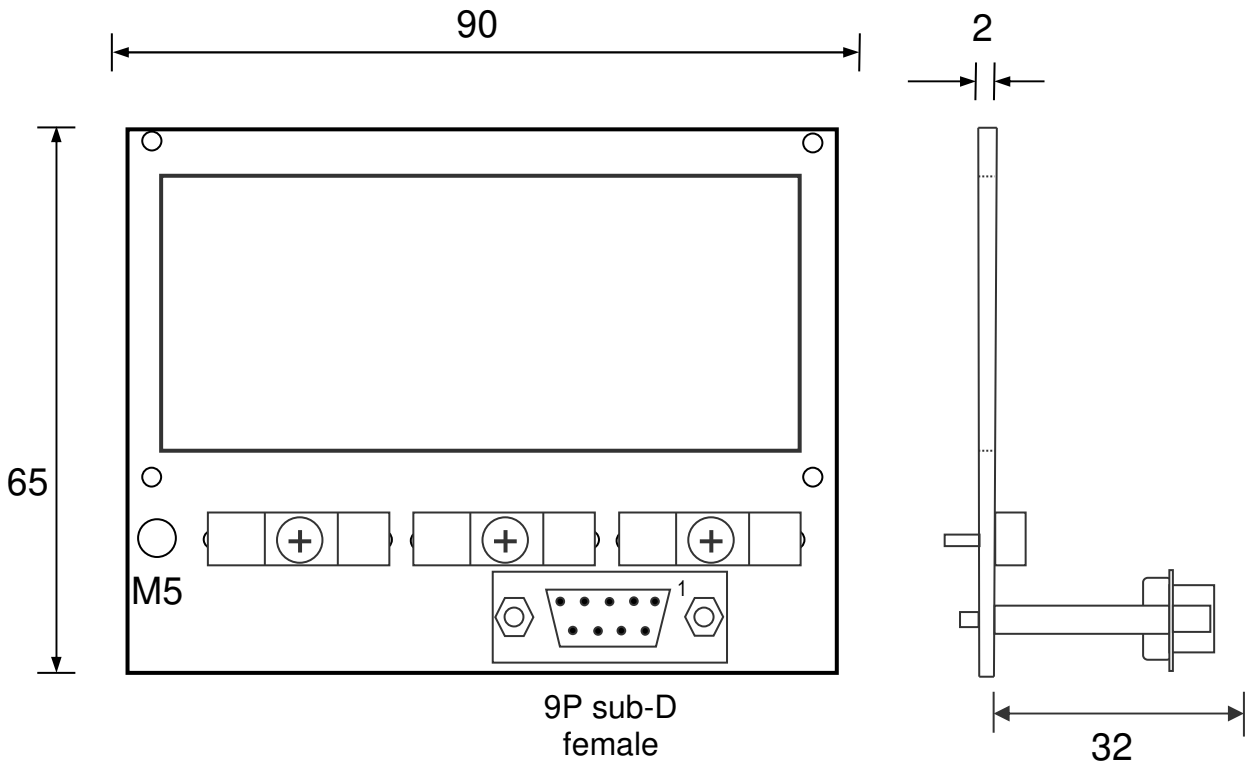


Top view

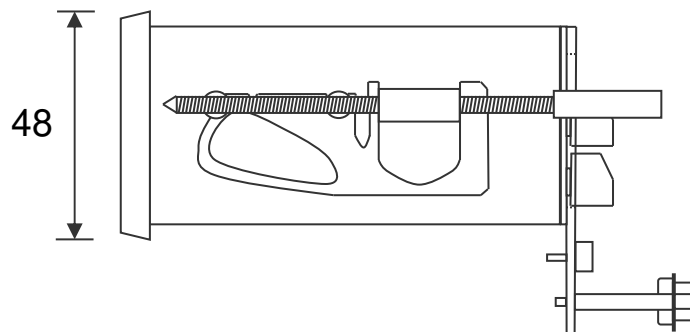


7.4 Dimensions EMC bracket type EMC-B02

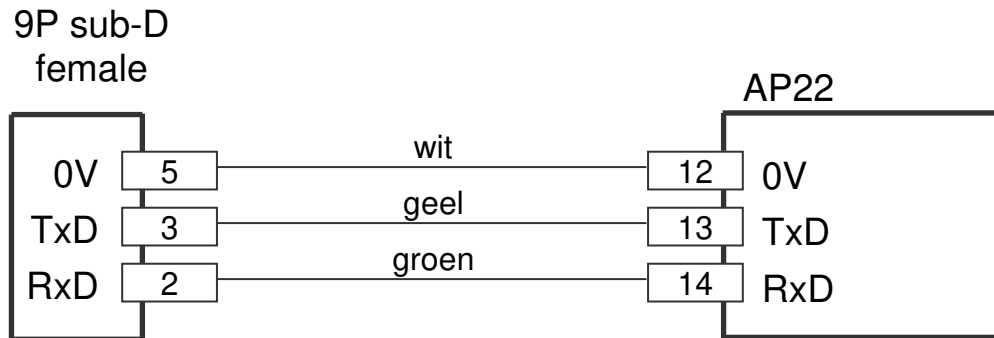
With RS232 (9P sub-D) on EMC bracket



Side view with EMC bracket



7.4.1 Connections RS232 on 9P sub-D connector



7.5 Dimensions protective hood type CDS-B02

