

Manual AP21 SSI Display



- 8 Digit display in DIN-enclosure 96 X 48 mm
- Supply voltage 10 – 35V DC
- All in- and outputs optically isolated
- Input for absolute SSI encoders
- 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 analog output (16 bit) (with teach function)
- Programmable Linearization function

Table of contents

1	Introduction.....	5
1.1	GENERAL.....	5
1.2	IMPORTANT INFORMATION	6
1.3	EMC	6
1.4	DEFINITIONS.....	6
1.4.1	<i>Display units AWE</i>	6
1.4.2	<i>Parameter number</i>	6
1.4.3	<i>Notation</i>	7
1.4.4	<i>Edges</i>	7
2	Operation.....	8
2.1	KEY FUNCTIONS	8
2.2	KEY FUNCTIONS IN PROGRAMMING MODE	9
2.3	DISPLAY FUNCTIONS.....	10
2.3.1	<i>Status functions</i>	10
2.3.2	<i>Error messages</i>	10
2.3.3	<i>Survey of error messages</i>	11
3	Programming	12
3.1	AUTOMATIC MODE	12
3.1.1	<i>Monitor function</i>	13
3.1.2	<i>Displaying the type number</i>	14
3.1.3	<i>Displaying the software-version</i>	14
3.1.4	<i>Status in- and outputs</i>	15
3.2	CHANGING NOMINAL VALUES.....	16
3.3	CHANGING PARAMETERS	17
3.3.1	<i>Menus</i>	17
3.3.2	<i>Input parameters</i>	18
4	Functions.....	19
4.1	ACTUAL POSITION	19
4.2	VELOCITY MEASUREMENT.....	19
4.3	MULTIPLICATOR.....	20
4.4	NUMBER OF BITS SSI.....	21
4.5	SSI MONITORING	22
4.5.1	<i>Output "SSI error"</i>	23
4.5.2	<i>Reset "SSI error"</i>	23
4.6	JUSTAGE SSI	23
4.6.1	<i>Preset using the parameter "Adjustment absolute value"</i>	23
4.6.2	<i>Using the input (input 1 or 2)</i>	24
4.6.3	<i>Using service parameter P[207]</i>	24
4.6.4	<i>Justage using keys</i>	24
4.7	ASCII PROTOCOL	25
4.7.1	<i>Overview functions</i>	25
4.7.2	<i>General</i>	26
4.7.3	<i>Functions</i>	27
4.7.4	<i>Error messages</i>	32
4.8	ANALOG OUTPUT.....	33
4.8.1	<i>Voltage output</i>	33
4.8.2	<i>Current output</i>	34

4.8.3	<i>Example programming voltage output</i>	35
4.9	TEACH FUNCTION ANALOG OUTPUT.....	35
4.10	CAMS.....	36
4.10.1	<i>General</i>	36
4.10.2	<i>Switching type (1) Cam with start- and end-value</i>	37
4.10.3	<i>Switching type (2) Greater than or equal to limit value</i>	37
4.10.4	<i>Switching type (3) Smaller than or equal to limit value</i>	37
4.10.5	<i>Switching type (4) Greater than or equal to limit value (Timer)</i>	38
4.10.6	<i>Switching type (5) Smaller than or equal to limit value (Timer)</i>	38
4.10.7	<i>Dynamic cams</i>	39
4.10.8	<i>Start/stop cam</i>	40
4.10.9	<i>Output "Cams active"</i>	40
4.11	LINEARIZATION	41
4.11.1	<i>mode 0 "4-Quadrant"</i>	42
4.11.2	<i>mode 1 "Mirror Y-axis"</i>	43
4.11.3	<i>mode 2 "Mirror XY-axis"</i>	43
5	Parameters	44
5.1	MENU 1 CONFIG	44
5.2	MENU 2 ACTUAL	46
5.3	MENU 3 SERIAL	48
5.4	MENU 4 INPUT	49
5.5	MENU 5 OUTPUT.....	50
5.5.1	<i>Submenu 5.1 – 5.4 Op1...4</i>	50
5.6	MENU 6 CAM.....	51
5.6.1	<i>Submenu 6.1 ... 6.12 CA1...12</i>	51
5.7	MENU 7 ANALOG OUTPUT	52
5.7.1	<i>Submenu 7.1 Config</i>	52
5.7.2	<i>Submenu 7.2 DA-U (voltage)</i>	52
5.7.3	<i>Submenu 7.3 DA-I (current)</i>	53
5.8	MENU 8 LINEARIZATION.....	54
5.8.1	<i>Submenu 8.1 Config</i>	54
5.8.2	<i>Submenu 11.2 ... 11.31 P1...P30</i>	55
5.9	OVERVIEW PARAMETERS.....	56
6	Connections	58
6.1	OVERVIEW CLAMP CONNECTIONS.....	58
6.2	SUPPLY.....	59
6.3	SSI INPUT	60
6.4	SSI ENCODER 24V	60
6.5	DIGITAL INPUTS.....	61
6.6	DIGITAL OUTPUTS.....	61
6.7	RS232	62
6.8	ANALOG OUTPUT.....	62
7	Technical specifications	63
7.1	SPECIFICATIONS	63
7.2	TYPEKEY	65
7.3	DIMENSIONS AP21	66
7.4	DIMENSIONS EMC BRACKET TYPE EMC-B02.....	67
7.4.1	<i>Connections RS232 on 9P sub-D connector</i>	68
7.5	DIMENSIONS PROTECTIVE HOOD TYPE CDS-B02.....	69

1 Introduction

1.1 General

With the microcontroller based AP21 it is possible to connect absolute sensors with SSI interface (Synchron Serial Interface)

It is standard equipped with 2 digital inputs and 4 digital outputs, all free programmable. The AP21 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 AP21 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 AP21 is configurable for numerous applications by adjusting its parameters.

The AP21 can be programmed with the DST2 PC-based software.

1.2 Important information

- ✓ The AP21 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 AP21.
- ✓ If through a failure or fault of the AP21 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 AP21.
- ✓ Necessary repairs to the AP21 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 AP21:

- ✓ 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)

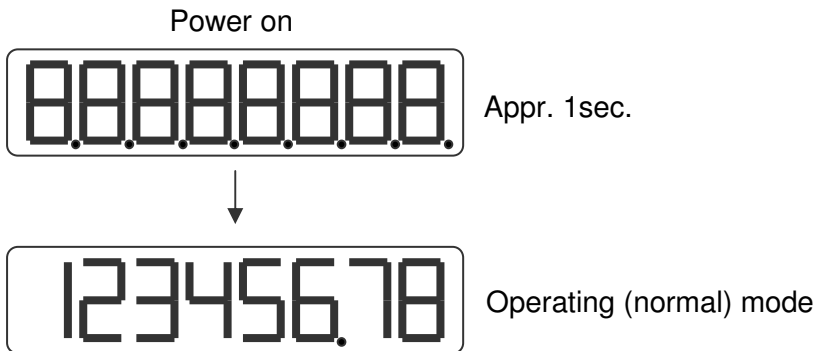


[Enter] key

- 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

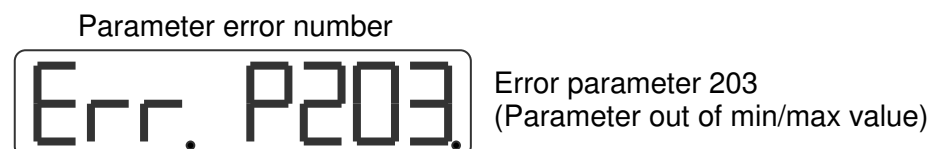


2.3.2 Error messages

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.

701 = Adjustment value SSI P[005] \geq Counting range
or Adjustment SSI P[005] $<$ Offset P[002]

703 = Number of active SSI databits $>$ number of SSI Clock pulses

705 = Offset SSI P[002] $>$ Counting range

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)

800 = SSI error delta-s

801 = SSI error cable failure

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

9xx = Linearization $X_n \leq X_{n-1}$

940 = Linearization (mode 1 or 2) $X_1 \neq 0$

941 = Linearization mode 1) $Y_1 \neq 0$

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 to 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

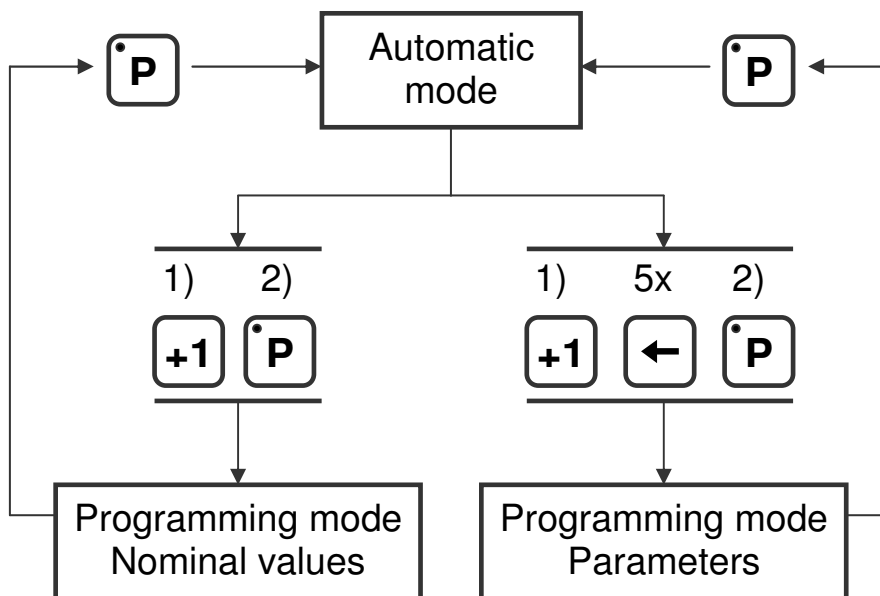
er 6 = data invalid (outside min/max value)

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

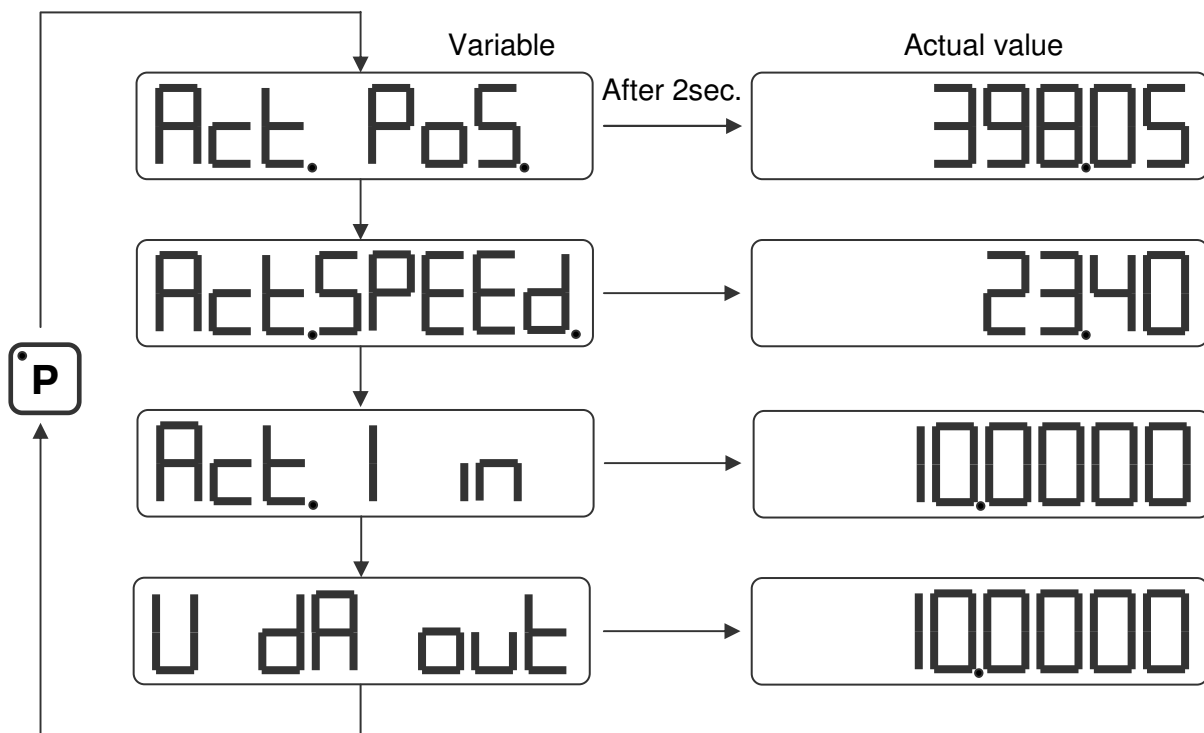
3.1 Automatic mode

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

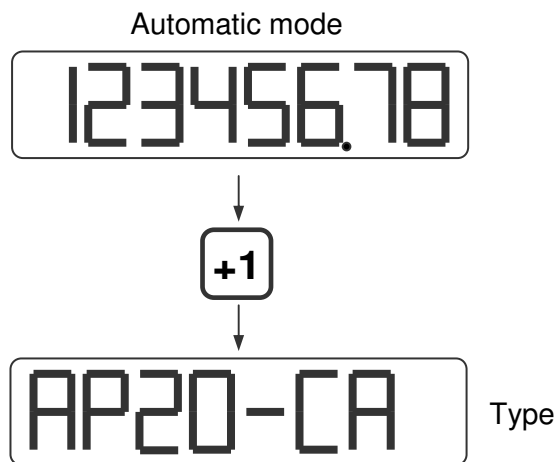
In automatic mode different variables can be displayed. By using the [P] key one can cycle through the different pages:

- Actual position
- Actual velocity
- Actual linearization
- Voltage or current of the DA (analog out)

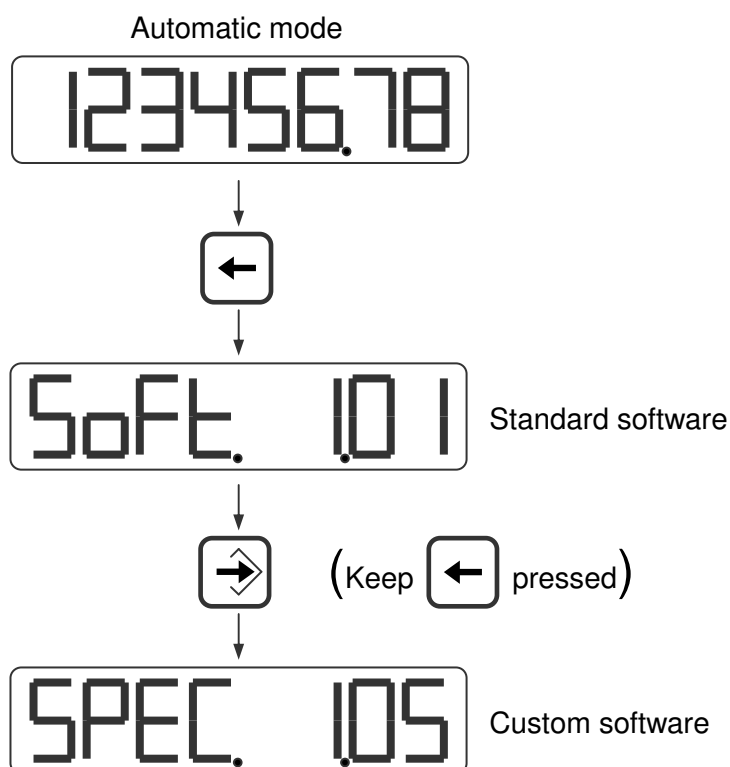


Parameter P[208] determines which option is visible after start-up.

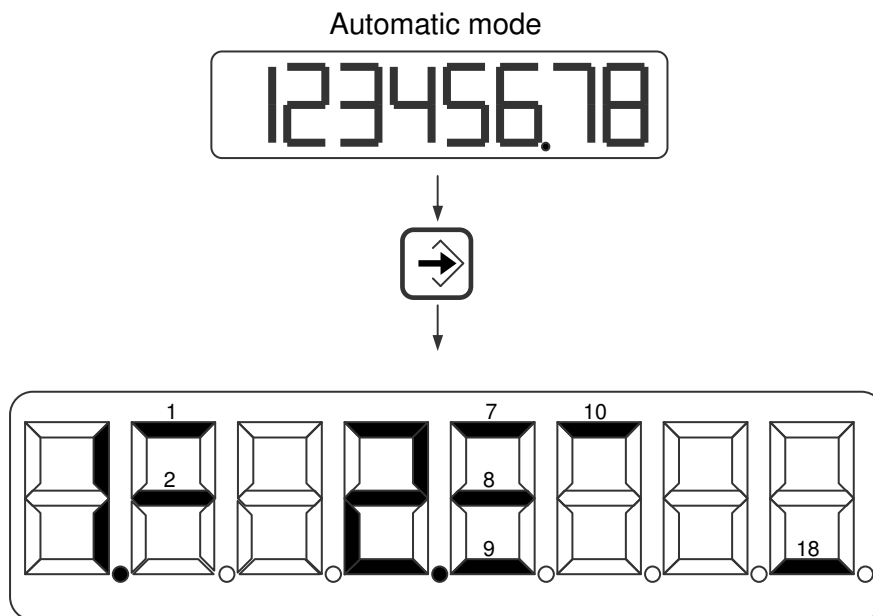
3.1.2 *Displaying the type number*



3.1.3 *Displaying the software-version*



3.1.4 Status in- and outputs



Inputs

1 = input-1
2 = input-2

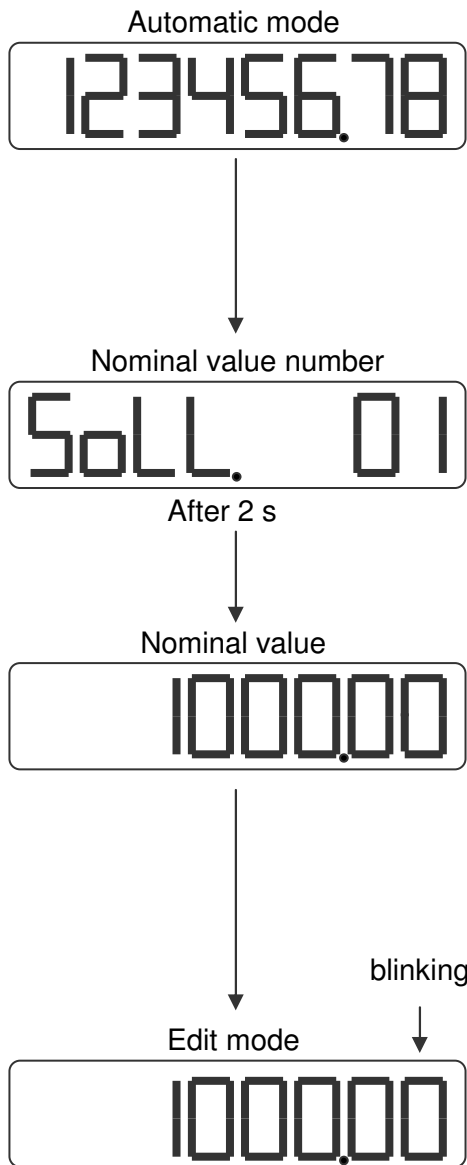
Outputs

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

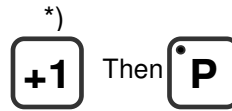
Outputs

18 = Cams active

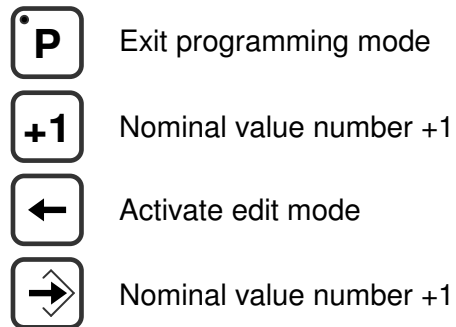
3.2 Changing nominal values



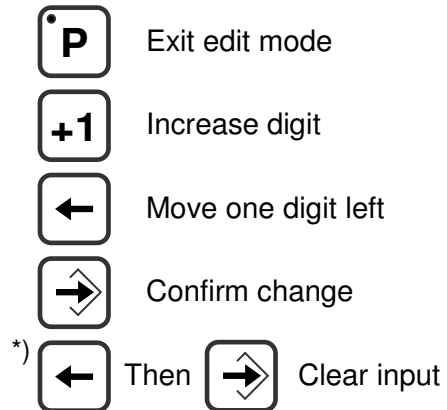
Access to nominal values



Select nominal values



Changing values



*) keep pressed

3.3 Changing parameters

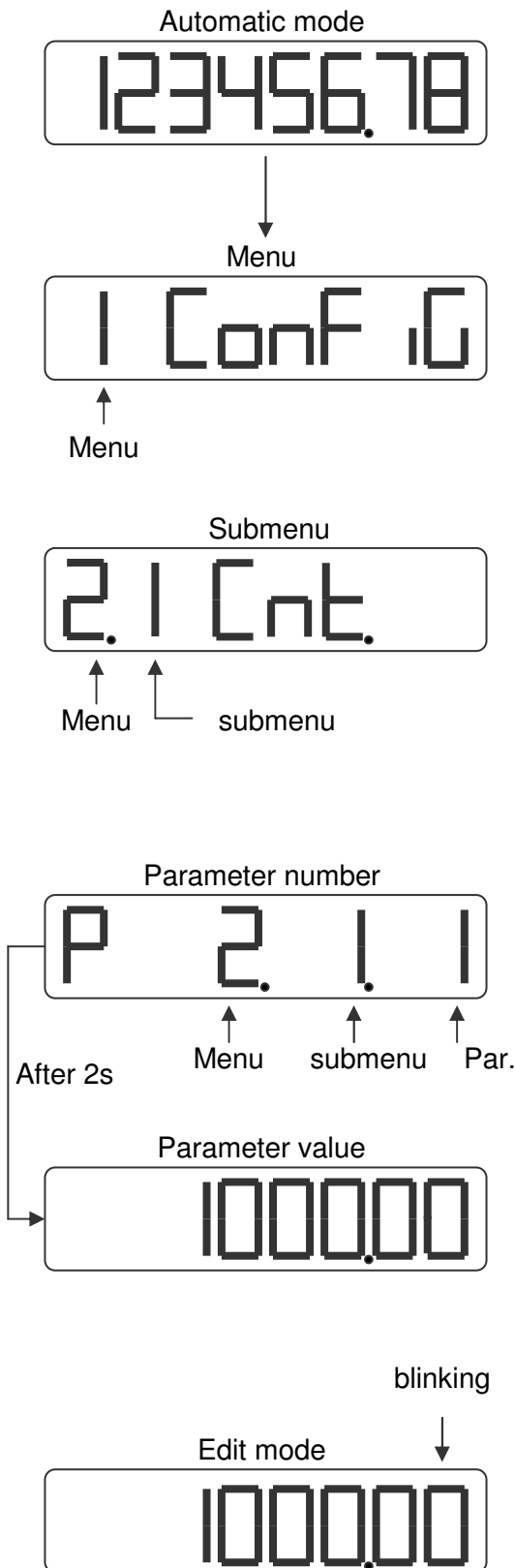
3.3.1 Menus

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
 - 7.1 ConFG.
 - 7.2 dA-U
 - 7.3 dA-I
- 8 Lin
 - 8.1 ConFG.
 - 8.2 P 1
 -
 -
 - 8.31 P30

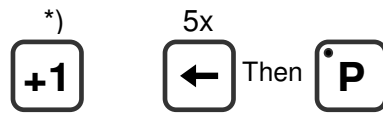
Example:
PAr. 6.1.1 means
menu 6, submenu , parameter 1

3.3.2 Input parameters

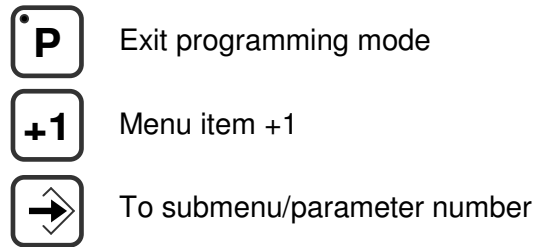


*) keep pressed down

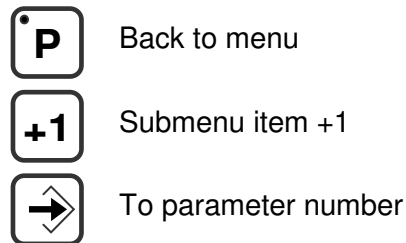
Access parameters



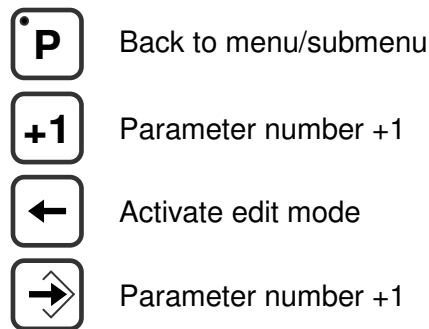
Menu selection



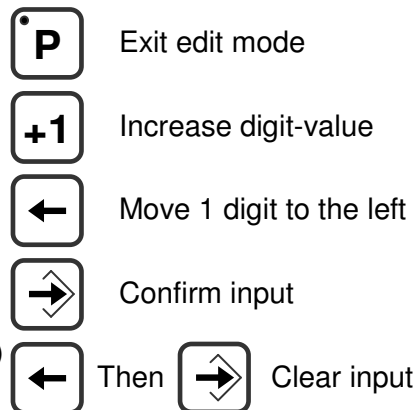
Submenu selection



Selecting parameters



Changing parameters



4 Functions

4.1 Actual position

The actual position shown on the display.

$$\text{Actual position} = \text{SSI} \times \text{dir} \times \frac{\text{Mt}}{\text{Mn}} + \text{N}$$

Mt	=	multiplicator numerator	P[000]
Mn	=	multiplicator denominator	P[001]
N	=	offset	P[002]
dir	=	direction (x1 or x -1)	P[211]

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 *Multiplicator*

By using the multiplicator it is possible to convert the SSI value to display-units (AWE).

Example:

Encoder with 90° shifted signal and 1024 increments/revolution will have 4096 edges/revolution. If the desired value in the display is 360,0 (= 3600 AWE), the multiplicator will be $3600/4096 = 0,87890625$.

It is possible to program the exact values (3600 and 4096) instead of the fraction.

Multiplicator (numerator) P[000] = 3600

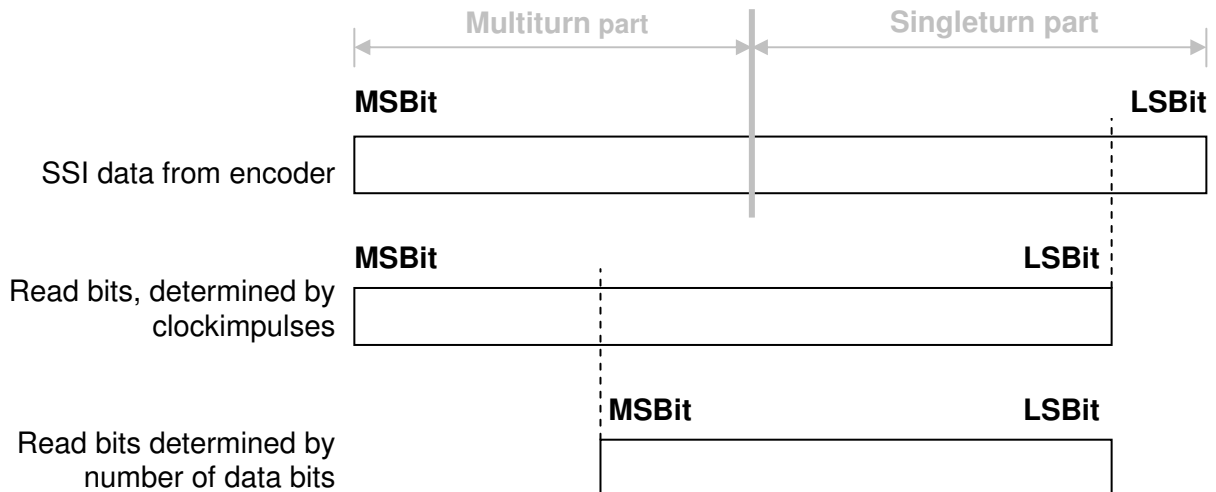
Multiplicator (denominator) P[001] = 4096

Through P[203] it is possible to adjust the decimal point.

4.4 Number of bits SSI

To read the sensor values of SSI-encoders two parameters are important:

Number of clock impulses P[216]
 Number of data bits P[217]



Normally the most significant bit (MSB) is transmitted first by the encoder.

The number of clock pulses determine the number of bits that will be read by the AP21. Basically this will be the number of bits that the encoder transmits.

Example:

The SSI-encoder has a resolution of 4096 positions per revolution and 4096 coded revolutions. The number of clock pulses will be 12 (bit) + 12 (bit) = 24 (bit).

The number of data bits will normally be the same: 24.

In some cases however it can be desirable to change the number of clock pulses and/or data bits.

Example:

The SSI-encoder has 16 single turn bits (65535 positions/rev.) and 14 multi turn bits (16384 coded revolutions).

The resolution should be 8192 positions/rev. = 13 bit. The number of revolutions should not be changed (remains 14).

Number of clock pulses will be: $P[216] = 27 (13+14)$

Number of data bits will be: $P[217] = 27 (13+14)$

4.5 SSI monitoring

The SSI-value is being read and processed every single cycle (250µs). Due to external electromagnetic interference it may happen that during 1 cycle incorrect information has been read.

To prevent unwanted cam actions switching actions, the SSI value can be monitored.

There are two different ways of monitoring the SSI-value: detection of cable-failure (see also P[232], P[252]) and monitoring the delta-SSI.

The delta-SSI value is a maximum value that two separate readings may differ from each other (P[006]). It is possible to set the number of times that this value may be overrun (P[220]).

If there is a reading which is exceeding the max. difference, the last reading is interpolated. When the max overrun counting is reached there will be a SSI error generated.

4.5.1 Output "SSI error"

It is possible to create a SSI-error signal using one of the outputs 1..4:

Output SSI error

Output-x = option "**2 SSI error**" (Low = SSI error)

4.5.2 Reset "SSI error"

The SSI error-signal can be reset by using one of the following options:

- Activate and exit programming mode parameters
- Reset with PC-program DST2
- Serial communication (ASCII)
- Set one of the inputs 1...2 to "**4 Reset SSI error**" and apply a high signal

4.6 Justage SSI

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

4.6.1 Preset using the parameter "Adjustment absolute value"

This function is applicable for absolute sensors and is active when parameter "Adjustment" P[218] 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.

$$\text{Display counting range} = \frac{\text{max SSI value} * \text{Multiplier (numerator) P[000]}}{\text{Multiplier (denominator) P[001]}}$$

4.6.2 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[218] 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[218]. 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.

4.6.3 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.

4.6.4 Justage using 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[218] = variant "5 front keys"

4.7 *ASCII protocol*

The serial port of the AP21 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.

4.7.1 *Overview functions*

sc	select AP21
r0	Read actual position (AWE)
r1	Read actual velocity (AWE/s)
r2	Read actual voltage analog output (0,1mV units)
r3	Read actual current analog output (0,1 mA units)
r8	Lezen actuele linearization (AWE)
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

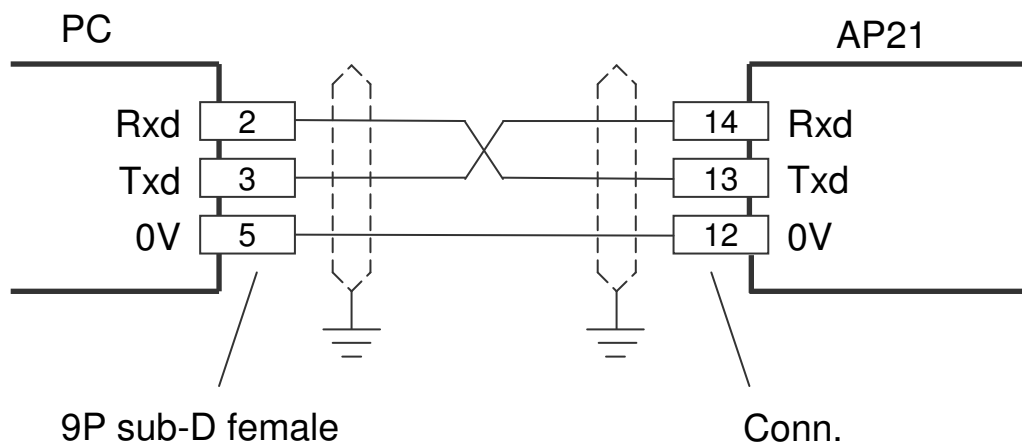
4.7.2 General

Through the ASCII protocol it is possible to communicate with the AP21.

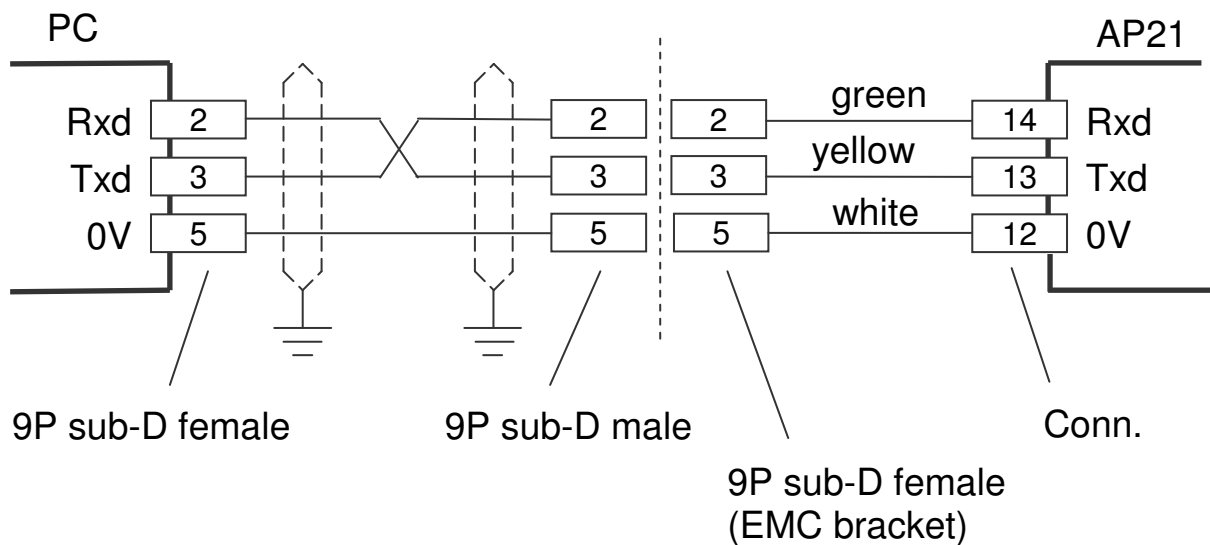
Send: Data from PC, PLC → AP21

Receive: Data from AP21 → 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)

4.7.3 Functions

sc Select AP21

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

The AP21 with the unit Id number is selected, all consecutive commands are relevant for this unit.

An AP21 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

r2 Read actual voltage analog output (0,1mV units)

send: r2
receive: r2 xxxxxxxx
transmitting parameter: none

r3 Read actual current analog output (0,1 mA units)

send: r3
receive: r3 xxxxxxxx
transmitting parameter: none

r8 Lezen actuele linearization (AWE)

zenden: r8
ontvangen: r8 xxxxxxxx
overgaveparameter: geen

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 xxxxxxxx**

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 AP21-DA

transmitting parameter: none

Example:

send: rt

answer: rt AP21-DA

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)*

4.7.4 Error messages

In case of an error the AP21 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

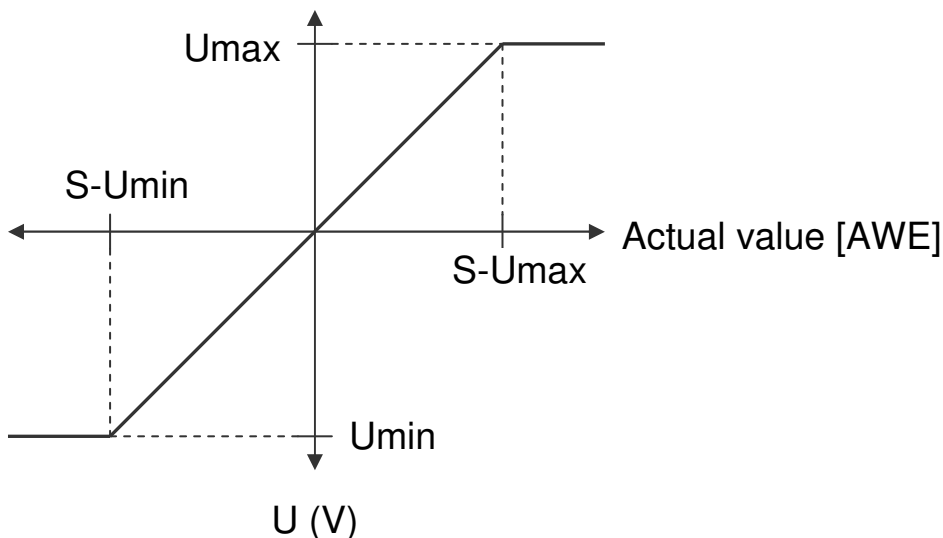
4.8 Analog output

The AP21 has an optional, galvanically isolated analog output. Using parameter P[231] it is possible to choose between a current output or a voltage output. The analog output can be used to give out the actual position, velocity or linearization (see parameter P[230]).

4.8.1 Voltage output

The voltage output has a resolution of 305 μV and is programmable through P[080] ... P[083].

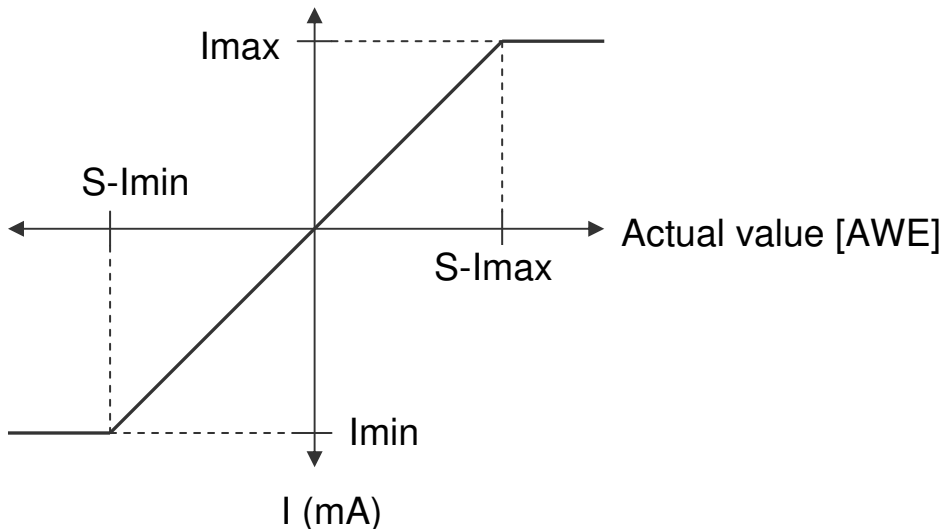
- P[080] = U_{min} [V] (input in 0,0001V units)
- P[081] = U_{max} [V] (input in 0,0001V units)
- P[082] = S- U_{min} [AWE] (actual value at U_{min})
- P[083] = S- U_{max} [AWE] (actual value at U_{max})



4.8.2 Current output

The current output has a resolution of 610 μA and is programmable through P[084] ... P[087].

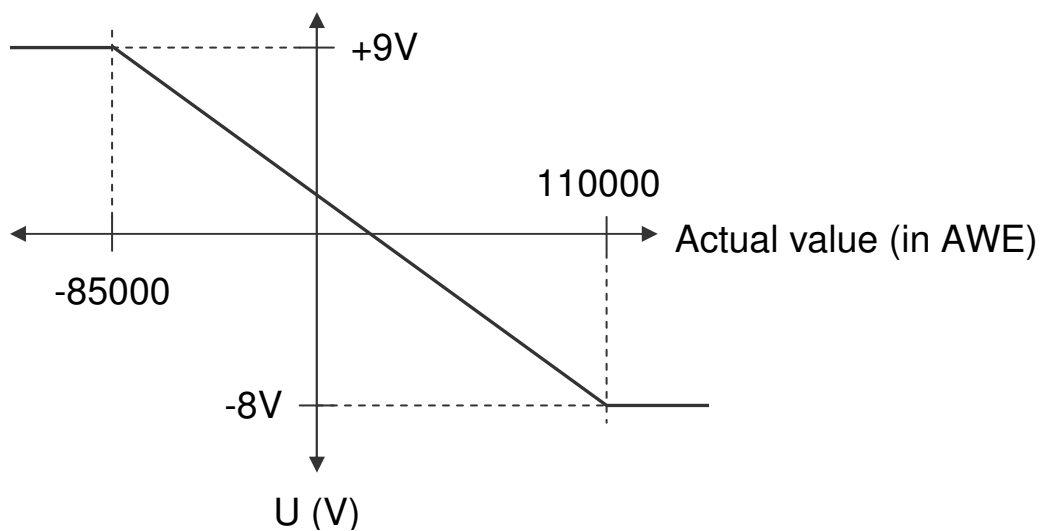
- P[084] = I_{min} [A] (input in 0,0001mA units)
- P[085] = I_{max} [A] (input in 0,0001mA units)
- P[086] = S- I_{min} [AWE] (actual value at I_{min})
- P[087] = S- I_{max} [AWE] (actual value at I_{max})



4.8.3 Example programming voltage output

- > 0,01 mm units
- > actual position at +9V should be -850,00 mm
- > actual position at -8V should be 1100,00 mm

P[080]	= Umin	= -8,0000 [V]
P[081]	= Umax	= +9,0000 [V]
P[082]	= S-Umin	= 110000 [AWE]
P[083]	= S-Umax	= -85000 [AWE]



4.9 Teach function analog output

The value for S_{min} and S_{max} can be set by an external input signal. For this function parameter P[234] and or P[235] can be set to variant "1 Teach via input". The function for the input 1...4 can be set to variant "11 DAC Set S_{min} " or variant "12 DAC Set S_{max} ".

On the positive edge of the input signal the teach function will be executed. The value of S_{min} or S_{max} will be stored with the actual position or the actual speed depending on parameter P[230]. The display will show "teach Lo" or "teach hi" for a short time.

4.10 Cams

4.10.1 General

The AP21 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)

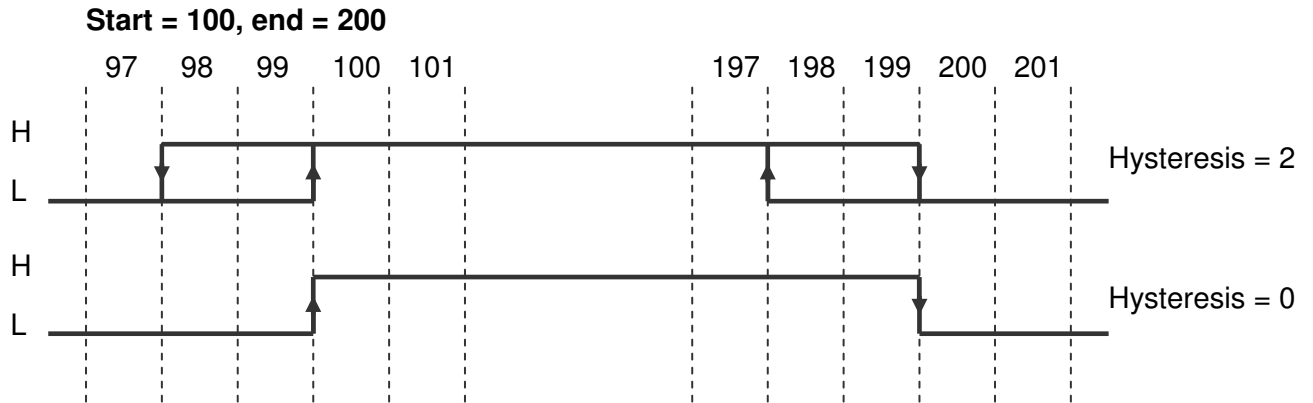
For switching type 1 there is a start- and end value. For Switching type 2...5 there is only a limit value needed.

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.

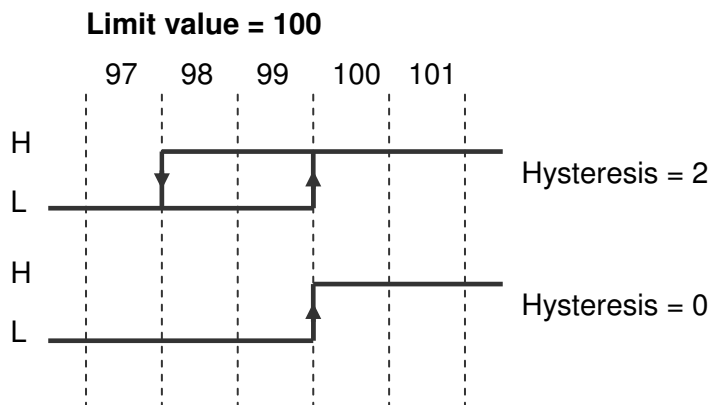
4.10.2 Switching type (1) Cam with start- and end-value

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



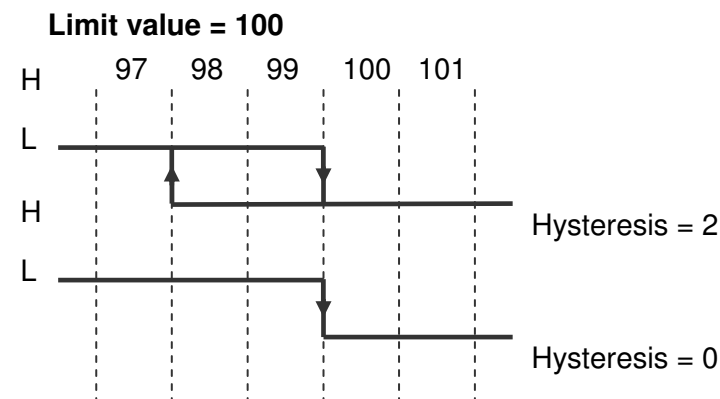
4.10.3 Switching type (2) Greater than or equal to limit value

Only one limit value needs to be programmed.



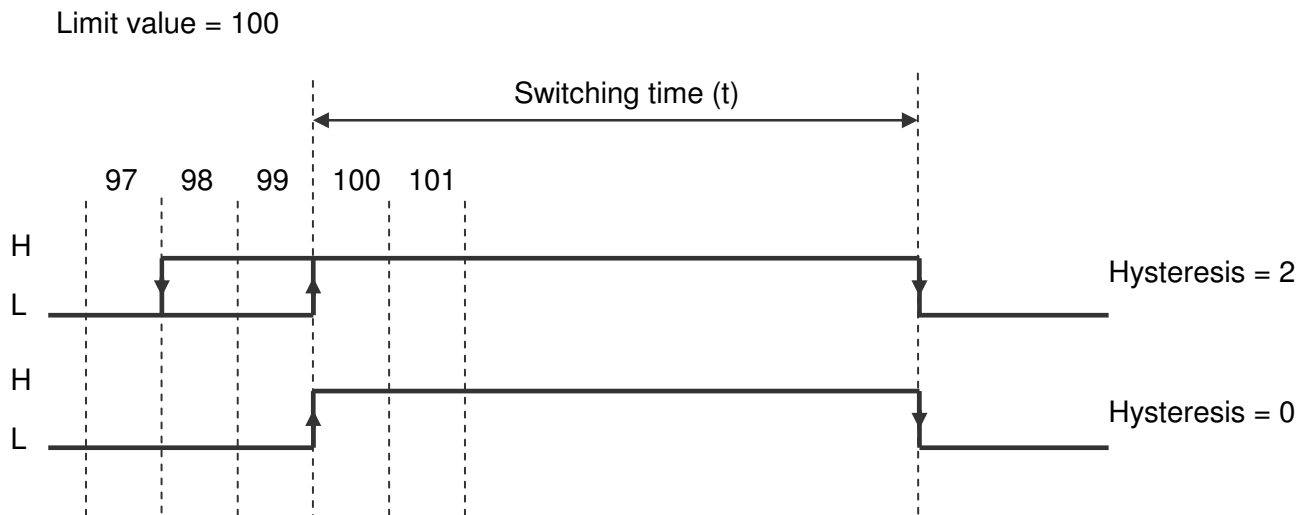
4.10.4 Switching type (3) Smaller than or equal to limit value

Only one limit value needs to be programmed.



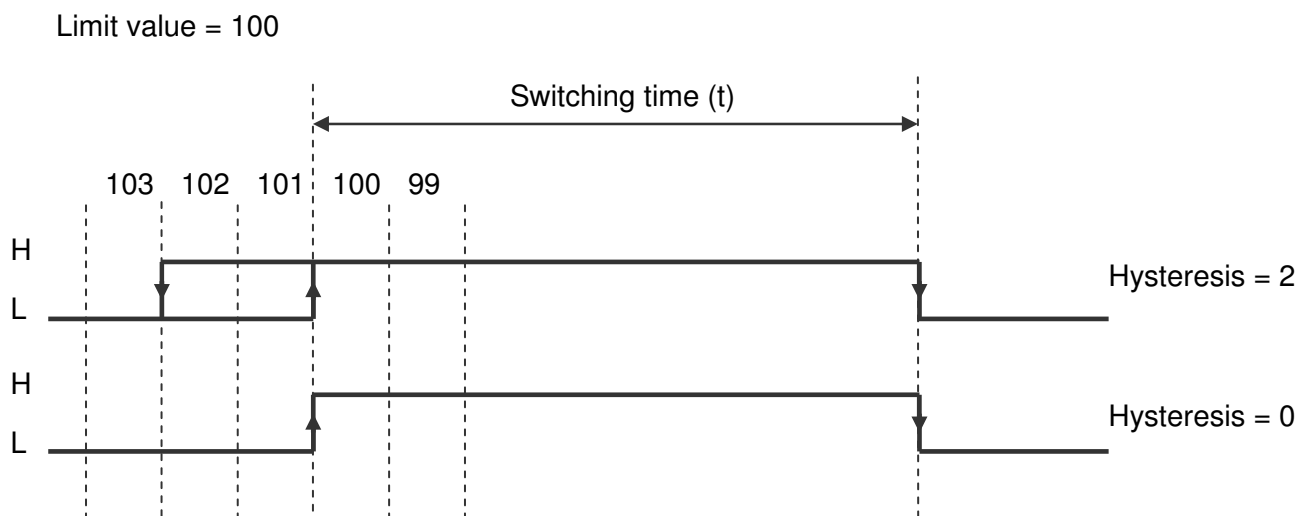
4.10.5 Switching type (4) Greater than or equal to limit value (Timer)

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.



4.10.6 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.



4.10.7 Dynamic cams

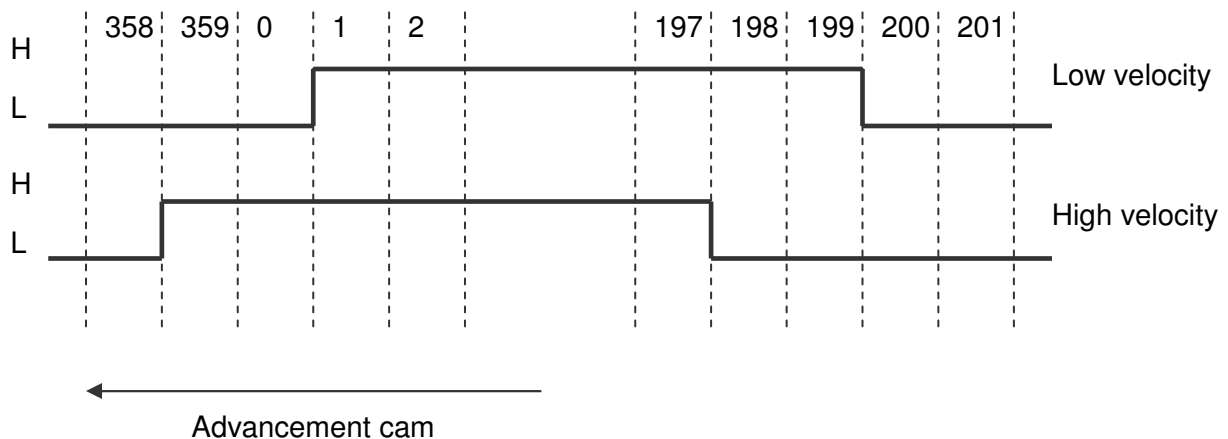
To compensate for actions with a static time, for example the switch time of a valve, it is possible to program a time for each output individually. The cams will be shifted according to this programmed time.

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:

<i>startvalue:</i>	<i>1</i>
<i>endvalue:</i>	<i>200</i>
<i>hysteresis:</i>	<i>0</i>
<i>counting range display:</i>	<i>360</i>



4.10.8 *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)

4.10.9 *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).

4.11 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"

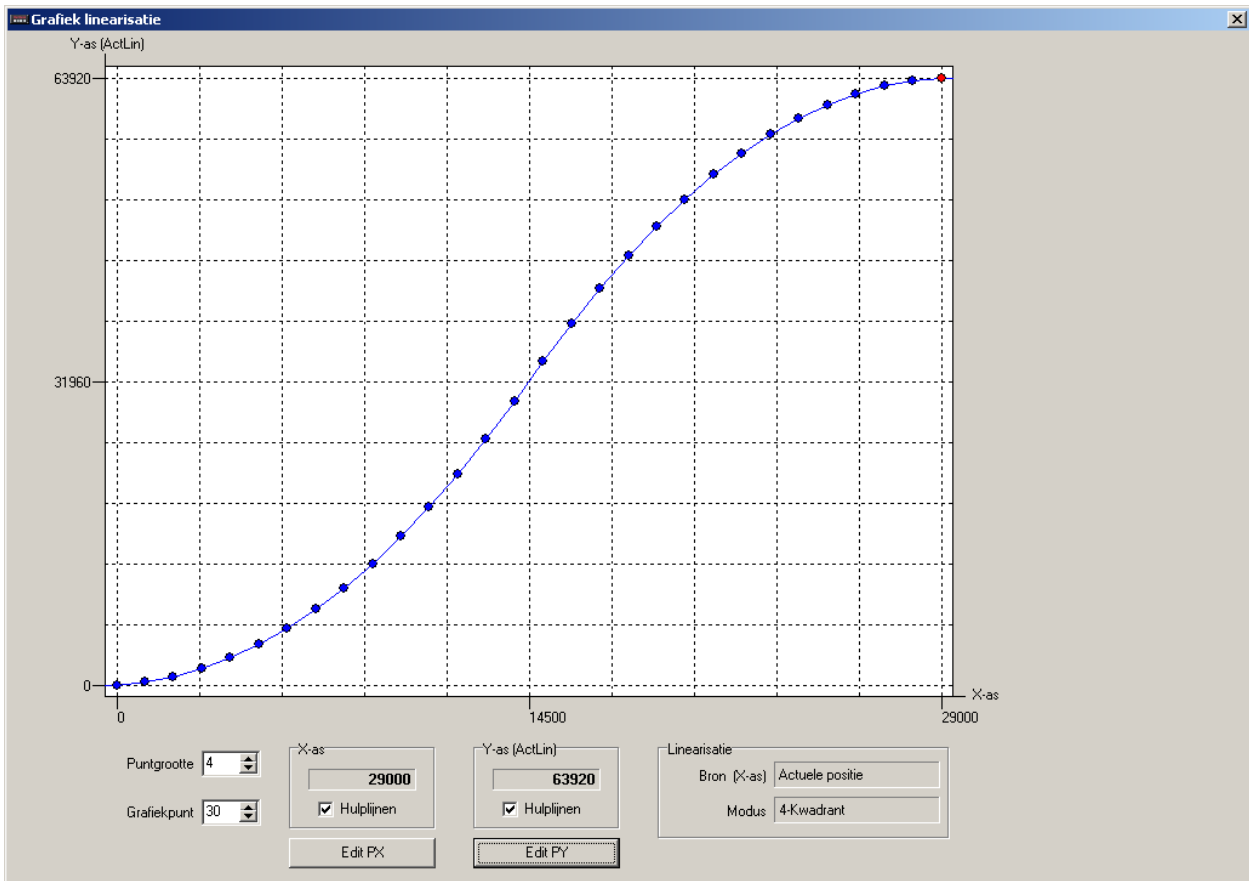
optie "2 Mirror XY-axis"

4.11.1 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:

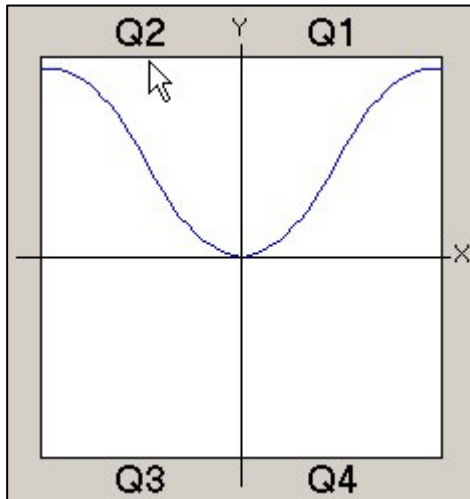
$P1-X < P2-X < P3X < \dots < P30-X$



Screenshot from PC programm DST2

4.11.2 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.



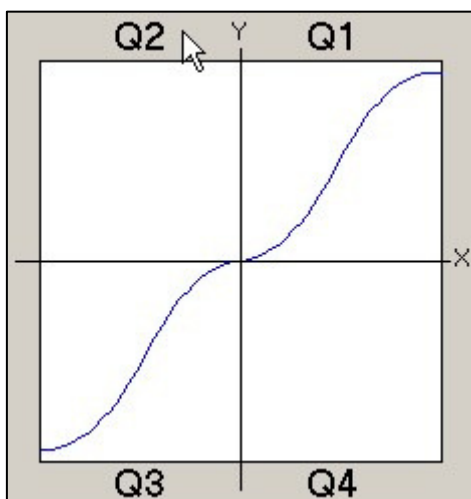
Conditions:

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

$$P1-X = 0$$

4.11.3 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:

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

$$P1-X = 0$$

$$P1-Y = 0$$

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 AP21		
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 ... 2
Default monitor function		
Determines the default which is visible after start-up.		
0 = Actual position		
1 = Actual velocity		
2 = Actual linearization		

5.2 Menu 2 Actual

PAR: 2.2.1	P[214]	0 ... 1
SSI code		
0 = gray		
1 = binary		

PAR: 2.2.2	P[211]	0 ... 1
Counting direction		
0 = no reverse		
1 = reverse		

PAR: 2.2.3	P[216]	0 ... 24 ... 30
Number of SSI clockpulses		
XX		

PAR: 2.2.4	P[217]	0 ... 24 ... 30
Number of SSI databits		
XX		

PAR: 2.2.5	P[000]	0 ... 10000 ... 16777215
Multiplier numerator		
XXXXXXXXXX		

PAR: 2.2.6	P[001]	0 ... 10000 ... 16777215
Multiplier denominator		
XXXXXXXXXX		

PAR: 2.2.7	P[218]	0 ... 6
Adjustment		
0 = no function		
1 = rising edge (input) only temporary in RAM (on interrupt)		
2 = falling edge (input) only temporary in RAM (on interrupt)		
3 = set with parameter adjustment value		
4 = rising edge (input) permanent in EEPROM (not on interrupt)		
5 = front keys		

PAR: 2.2.8	P[213]	0 ... 2
Reference coarse		
0 = no function		
1 = high signal		
2 = low signal		

PAR: 2.2.9	P[209]	0 ... 2
Counting direction adjustment		
0 = independant from direction		
1 = when counting in positive direction		
2 = when counting in negative direction		

PAR: 2.2.10	P[002]	-9999999 ... 0 ... 99999999
Offset		
-XXXXXXXX (AWE)		

PAR: 2.2.11	P[005]	-9999999 ... 0 ... 99999999
Adjustment value		
-XXXXXXXX (AWE) input 0 means function inactive		

PAR: 2.2.12	P[006]	1 ... 50 ... 99999
Monitoring delta-SSI per cycletime (250 µs)		
Unprocessed SSI-value, only depending on number of active SSI-databits P[217]		
XXXXX		

PAR: 2.2.13	P[220]	0 ... 2 ... 9
Maximum number of SSI errors to set external SSI-error. At every SSI-error the actual value is determined by interpolation of the last valid value		
X		

PAR: 2.2.14	P[221]	0 ... 3
SSI monitoring		
0 = not active		
1 = only wiring		
2 = only Delta SSI monitoring		
3 = wiring + Delta SSI monitoring		

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 ... 12
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		

INPUT-2

PAR: 4.0.2	P[250]	0 ... 12
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 >= limit value		
3 = actual position <= limit value		
4 = actual position >= limit value (Timer)		
5 = actual position <= limit value (Timer)		

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

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)		
-XXXXXXX		

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

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 7 Analog output

5.7.1 Submenu 7.1 Config

PAR: 7.1.1	P[231]	0 ... 2
Selection DA output		
0 = inactive		
1 = voltage		
2 = current		

PAR: 7.1.2	P[230]	0 ... 3
Selection DA source		
0 = actual position		
1 = actual velocity		
2 = actual velocity abs		
3 = actual linearization		

5.7.2 Submenu 7.2 DA-U (voltage)

DA PAR 7.2.1...7.2.4 = 0: DA not active

PAR: 7.2.1	P[080]	-100000 ... 99999
Umin DA		
-XX.XXXX (V)		

PAR: 7.2.2	P[081]	-99999 ... 100000
Umax DA		
-XX.XXXX (V)		

PAR: 7.2.3	P[082]	-9999999... -100000 ... 99999999
S-Umin DA		
-XXXXXXXX (AWE)		

PAR: 7.2.4	P[083]	-9999999 ... 100000 ... 99999999
S-Umax DA		
-XXXXXXXX (AWE)		

PAR: 7.2.5	P[234]	0 ... 1
Source for Smin		
0 = parameter Smin		
1 = teach by input		

PAR: 7.2.6	P[235]	0 ... 1
Source for Smax		
0 = parameter Smax		
1 = teach by input		

5.7.3 Submenu 7.3 DA-I (current)

DA PAR 7.3.1...7.3.4 = 0: DA not active

PAR: 7.3.1	P[084]	-200000 ... 199999
Imin DA		
-XX.XXXX (mA)		

PAR: 7.3.2	P[085]	-199999 ... 200000
Imax DA		
-XX.XXXX (mA)		

PAR: 7.3.3	P[086]	-9999999 ... -200000 ...99999999
S-Imin DA		
-XXXXXXXX (AWE)		

PAR: 7.3.4	P[087]	-9999999 ... 20000 ... 99999999
S-lmax DA		
-XXXXXXXX (AWE)		

PAR: 7.3.5	P[234]	0 ... 1
Source for Smin		
0 = parameter Smin		
1 = teach by input		

PAR: 7.3.6	P[235]	0 ... 1
Source for Smax		
0 = parameter Smax		
1 = teach by input		

5.8 Menu 8 Linearization

5.8.1 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		

5.8.2 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		

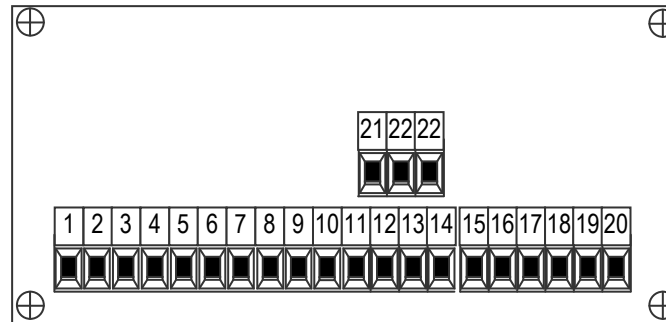
5.9 Overview parameters

<u>No</u>	<u>Description</u>	<u>Menu</u>
[000]	= Multiplier numerator	2.2.5
[001]	= Multiplier denominator	2.2.6
[002]	= Offset	2.2.10
[003]	= Reference value	2.1.8
[004]	= Counting range	2.1.9
[005]	= Adjustment value	2.2.11
[006]	= 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]...[079]	= no function	
[080]	= Umin DA	7.2.1
[081]	= Umax DA	7.2.2
[082]	= S-Umin DA	7.2.3
[083]	= S-Umax DA	7.2.4
[084]	= Imin DA	7.3.1
[085]	= Imax DA	7.3.2
[086]	= S-Imin DA	7.3.3
[087]	= S-Imax DA	7.3.4
[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]...[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], [201]	= no function	
[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	2.2.9
[210]	= no function	
[211]	= Counting direction	2.2.2

No	Description	Menu
[212]	= no function	
[213]	= Reference coarse	2.2.8
[214]	= SSI code	2.2.1
[216]	= Number of SSI clockpulses	2.2.3
[217]	= Number of SSI databits	2.2.4
[218]	= Adjustment (SSI)	2.2.7
[219]	= no function	
[220]	= Maximum number of SSI errors	2.2.13
[221]	= SSI monitoring	2.2.14
[222]...[229]	= no function	
[230]	= Selection DA source	1.1.2
[231]	= Selection DA output	7.1.1
[232], [233]	= no function	
[234]	= Source for Smin DA	7.2.5/7.3.5
[235]	= Source for Smax DA	7.2.6/7.3.6
[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]...[245]	= no function	
[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

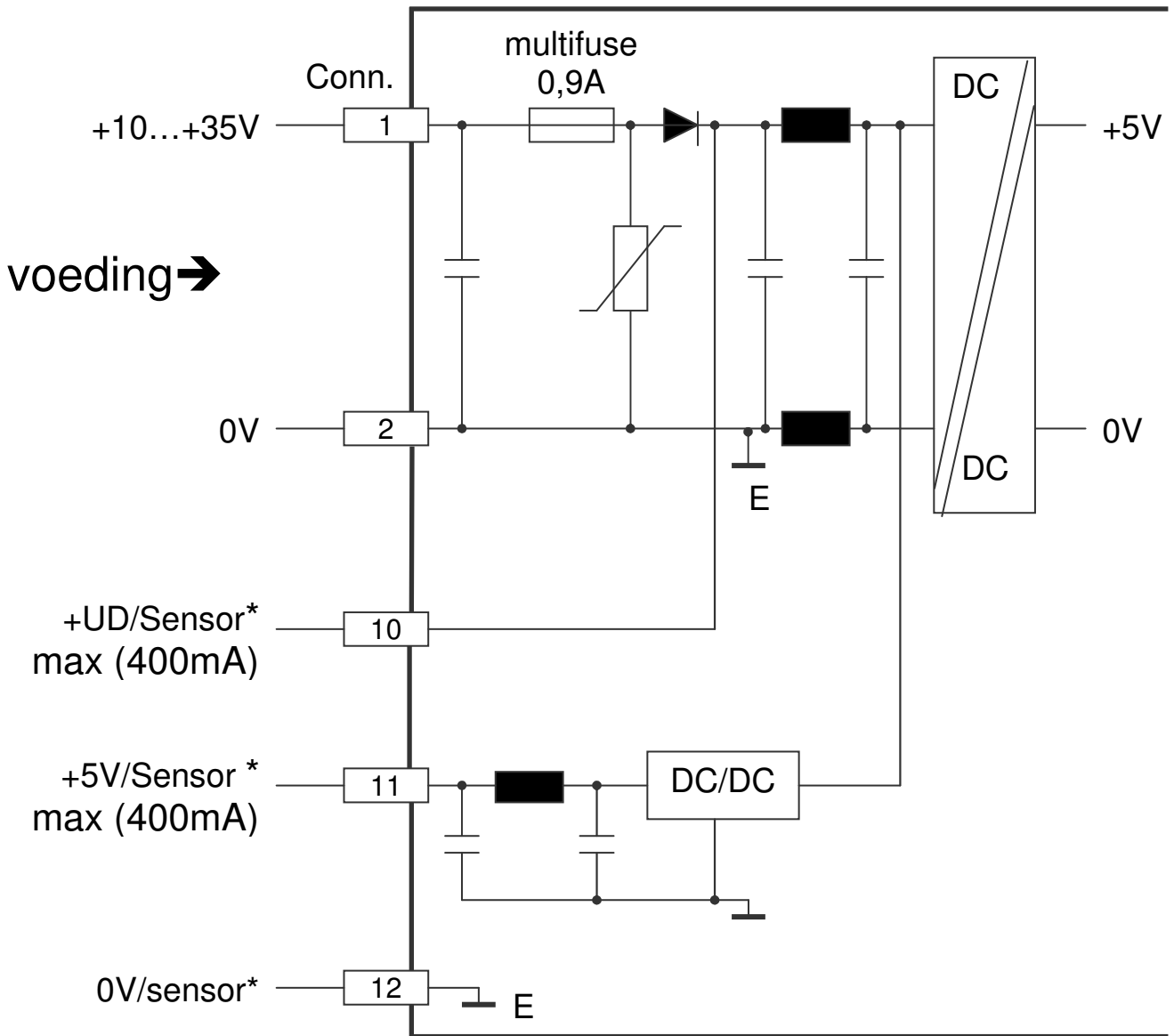


6.1 Overview clamp connections

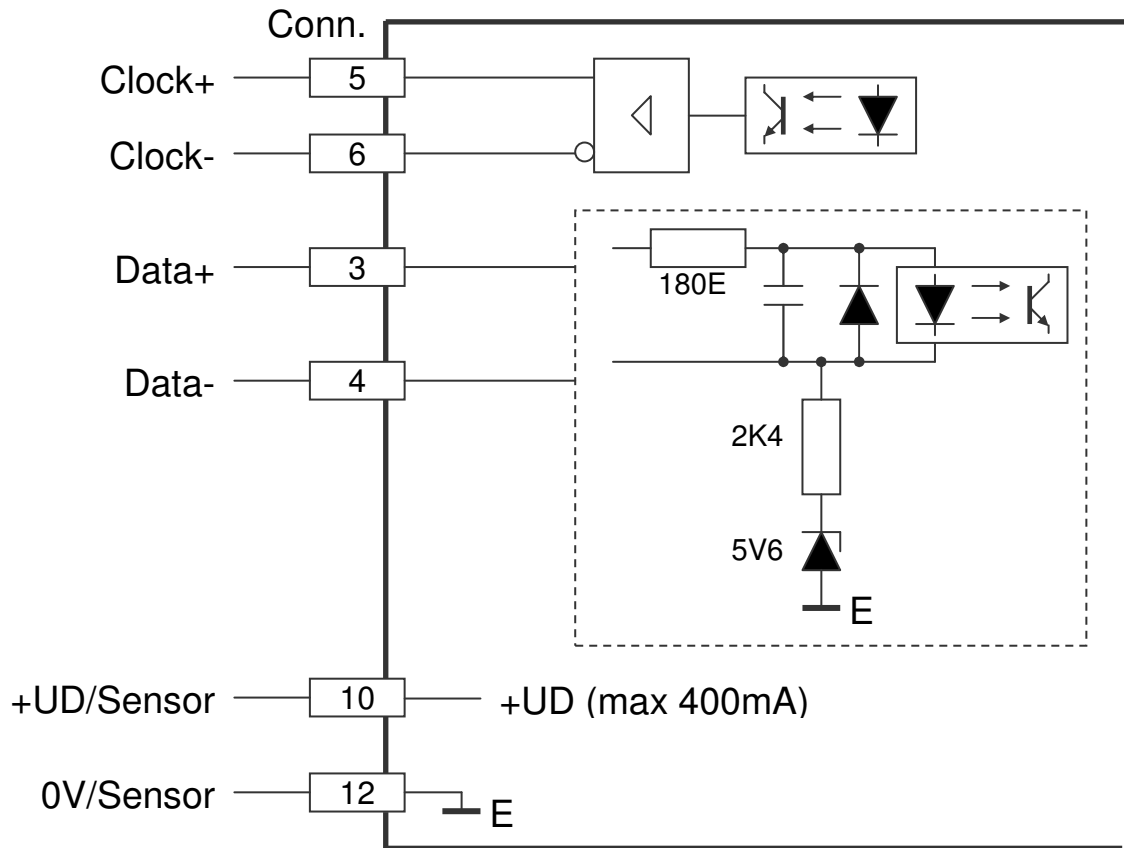
1. +10...+35V supply
2. 0V supply
3. Data+
4. Data-
5. Clock+
6. Clock-
7. Input-1
8. Input-2
9. common for inputs (0V)
10. +10...35V DC supply output for sensor
11. +5V DC supply output for sensor
12. 0V for sensor
13. TxD
14. RxD
15. Output -1
16. Output -2
17. Output -3
18. Output -4
19. 0V for outputs
20. +U for outputs

21. 0V analog
22. U-out
23. I-out

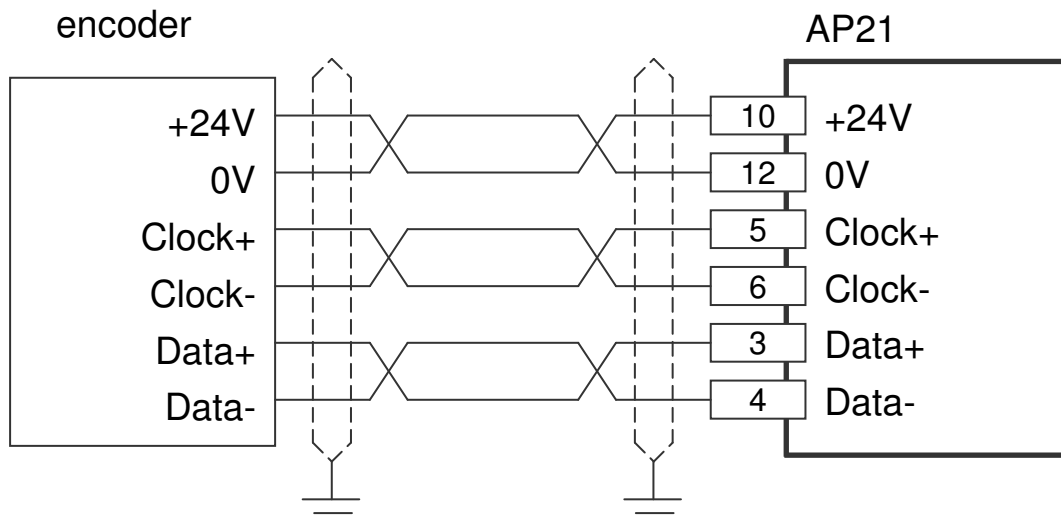
6.2 Supply



6.3 SSI input

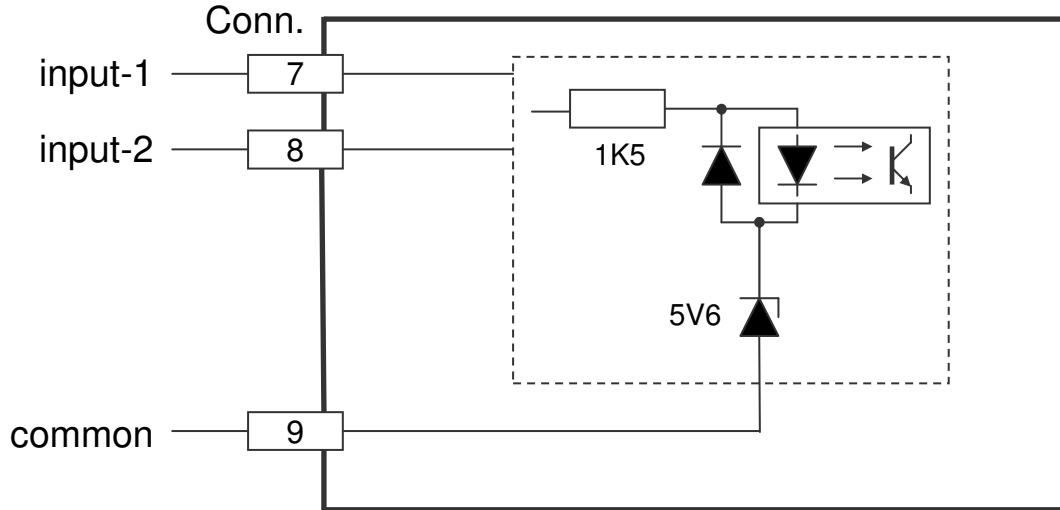


6.4 SSI encoder 24V

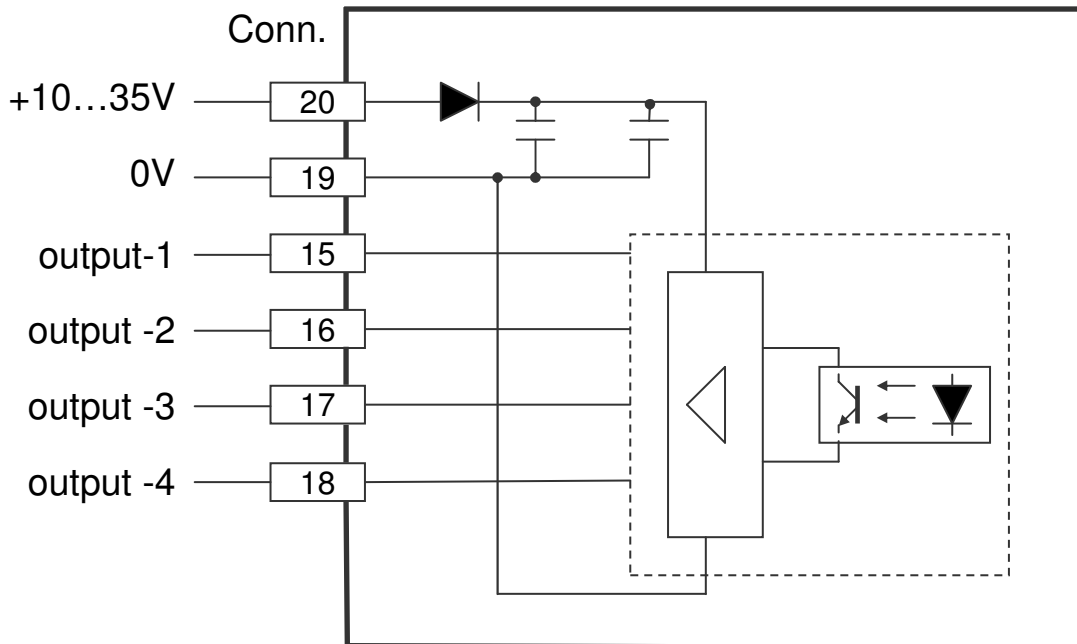


Supply voltage AP21 clamp 1 and 2 is 24V DC

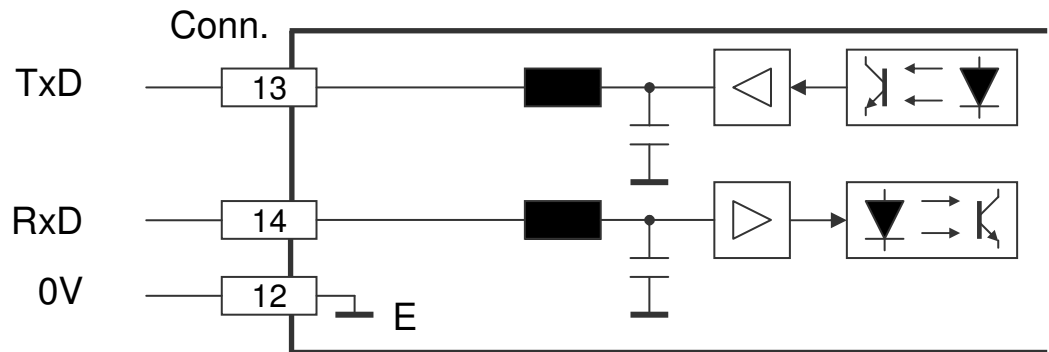
6.5 Digital inputs



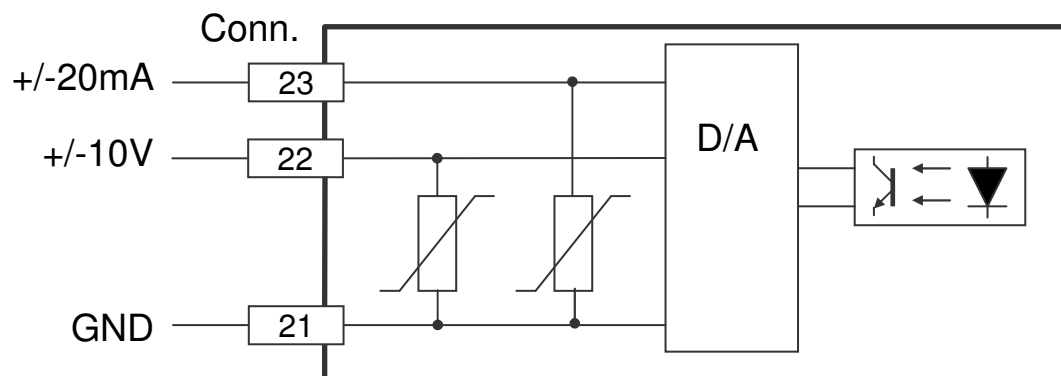
6.6 Digital outputs



6.7 RS232



6.8 Analog output



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
- SSI	optically isolated
data-input	low 0...+0,8V
	high +2,8V...+5V
clock-output	driver (RS422)
clock-frequency	125 kHz (138,9 kHz if > 26 bit encoder signal)
- 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 output	galvanically isolated
range	max. -10V ... +10V
resolution	305 μV
offset-temp. coeff.	< 20 ppm/ °C
I _{max}	+/-12mA

- Current output range
galvanically isolated
max. $-20\text{mA} \dots +20\text{mA}$
- resolution
 $610 \mu\text{A}$
- offset-temp. coeff.
 $< 20 \text{ ppm/ } ^\circ\text{C}$
- Rmax
 550 Ohm

- Serial communication
RS232 C

- Display
8 decades 7-segments LED
- digit height
 14 mm

- Temperature range
 $0 \dots 50 ^\circ\text{C}$

- Connection diameter
 $1,6 \text{ mm}^2$ (raster $3,81 \text{ mm}$)

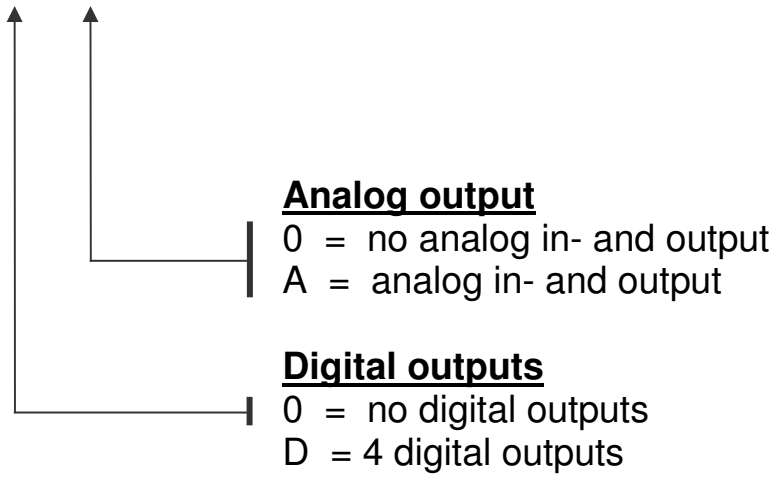
- Electromagnetic compatibility
in accordance with guideline
2004/108/EC
- emission
NEN-EN-IEC61000-6-3:2007
- immunity
NEN-EN-IEC61000-6-3:2005

- Weight
 $< 0.25 \text{ kg}$

- Sealing
front IP50, with protective hood IP54
rear IP20

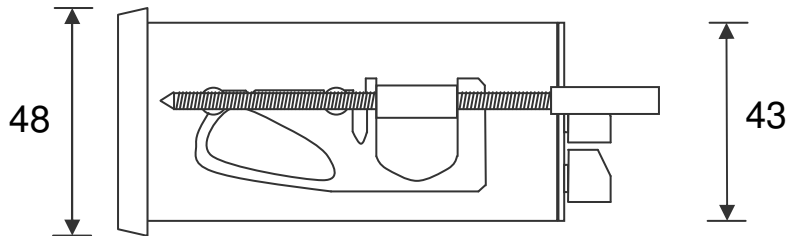
7.2 Typekey

AP21- X X

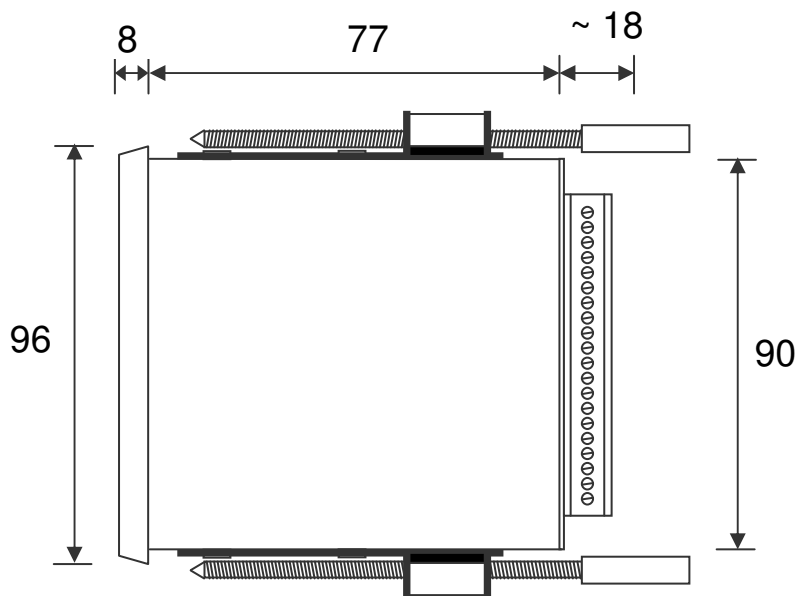


7.3 Dimensions AP21

Side view

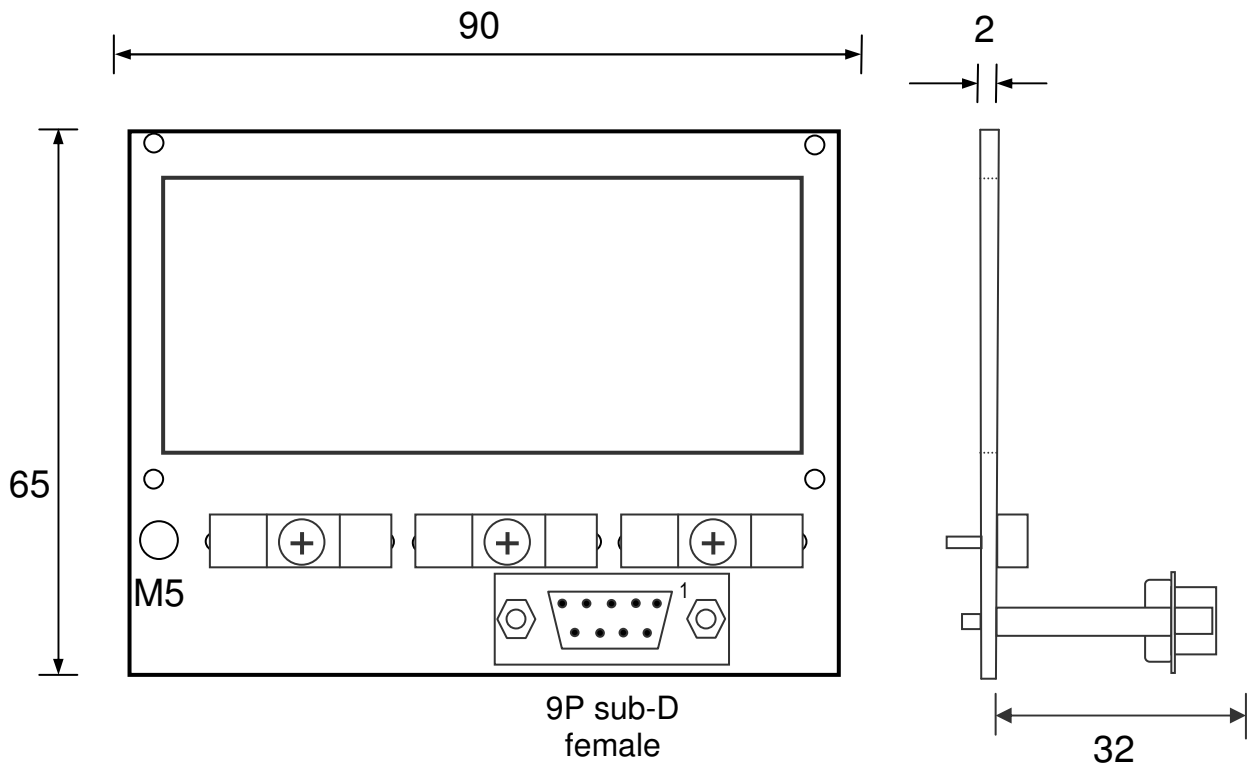


Top view

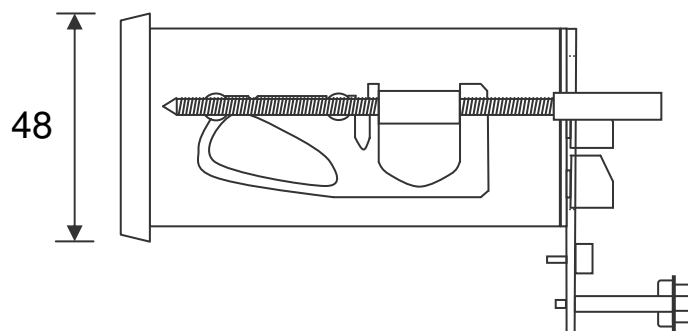


7.4 Dimensions EMC bracket type EMC-B02

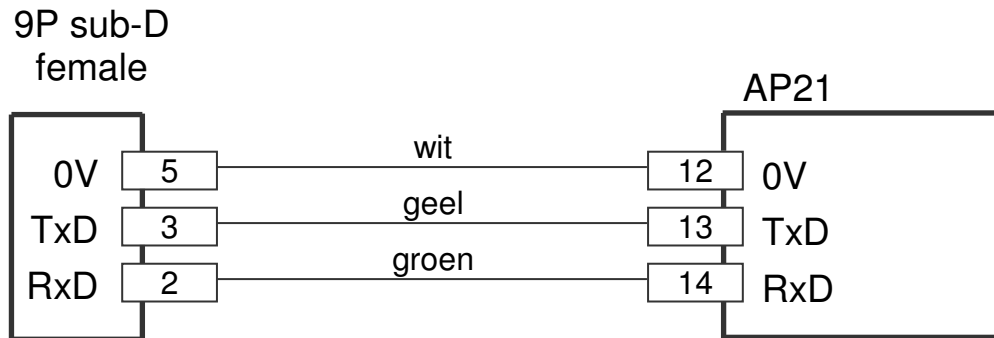
With RS232 (9P sub-D) on EMC bracket



Side view with EMC bracket



7.4.1 Connections RS232 on 9P sub-D connector



7.5 Dimensions protective hood type CDS-B02

