A
Pilz Ges.m.b.H.
Modecenterstraße 14
1030 Wien
Austria
Telephone (01) 7 98 62 63-0
Telefax (01) 7 98 62 64
E-Mail: pilz@pilz.at

AUS
Pilz Australia
Industrial Automation LP
9/475 Blackburn Road
Mt. Waverley, Melbourne VIC 3149
Australia
Telephone (03) 95 44 63 00
Telefax (03) 95 44 63 11
E-Mail: safety@pilz.com.au

B
Pilz Belgium
BC Building
L
Industriezone III
Industrielaan 4
9320 Erembodegem
Belgium
Telephone (0 53) 83 66 70
Telefax (0 53) 83 89 58
E-Mail: info@pilz.be

BR
Pilz do Brasil Sistemas Eletrônicos
Industriais Ltda.
Rua Ártico, 123 - Jd. do Mar
09726-300
São Bernardo do Campo - SP
Brazil
Telephone (11) 43 37-12 41
Telefax (11) 43 37-12 42
E-Mail: pilz@pilzbr.com.br

CH
Pilz Industrieelektronik GmbH
Gewerbepark Hintermättli
Postfach 6
5506 Mägenwil
Switzerland
Telephone (0 62) 8 89 79 30
Telefax (0 62) 8 89 79 40
E-Mail: pilz@pilz.ch

D
Headquarters:
Pilz GmbH & Co.
Felix-Wankel-Straße 2
73760 Ostfildern
Germany
Telephone (07 11) 34 09-0
Telefax (07 11) 34 09-1 33
E-Mail: pilz.gmbh@pilz.de

DK
Pilz Skandinavien KS
Ellegaardvej 25 L
6400 Sonderborg
Denmark
Telephone 74 43 63 32
Telefax 74 43 63 42
E-Mail: pilz@pilz.dk

E
Pilz Industrieelektronik S.L.
Edificio Tilma
Avda. Sant Julià 1
08400 Granollers
Spain
Telephone (93) 8 49 74 33
Telefax (93) 8 49 75 44
E-Mail: central@pilzspain.com

F
Pilz France Electronic
1, rue Jacob Mayer
BP 12
67037 Strasbourg Cedex
France
Telephone 03 88 10 40 00
Telefax 03 88 10 80 00
E-Mail: siege@pilz-france.fr

FIN
Pilz Skandinavien KS
Pakilantie 61
00660 Helsinki
Finland
Telephone (09) 27 09 37 00
Telefax (09) 27 09 37 09
E-Mail: pilz.sk@kolumbus.fi

GB
Pilz Automation Technology
Willow House, Medlicott Close
Oakley Hay Business Park
Corby
Northants NN18 9NF
United Kingdom
Telephone (0 15 36) 46 07 66
Telefax (0 15 36) 46 08 66
E-Mail: sales@pilz.co.uk

I
Pilz Italia Srl
Via Meda 2/A
22060 Novedrate (CO)
Italy
Telephone (0 31) 78 95 11
Telefax (0 31) 78 95 55
E-Mail: info@pilz.it

IRL
Pilz Ireland Industrial Automation
Cork Business and Technology Park
Model Farm Road
Cork
Ireland
Telephone (0 21) 4 34 65 35
Telefax (0 21) 4 80 49 94
E-Mail: sales@pilz.ie

J
Pilz Japan Co., Ltd.
Three One Building 701
3-20-5 Shin-Yokohama
Kohoku-ku
Yokohama 222-0033
Japan
Telephone (0 45) 4 71-22 81
Telefax (0 45) 4 71-22 83
E-Mail: pilz@pilz.co.jp

MEX
Pilz de Mexico S. de R.L. de C.V.
Av. San Ignacio 1079
Col. Jardines de San Ignacio
C.P. 45000
Guadalajara, Jalisco
Mexico
Telephone (0 13) 1 22 16 81
Telefax (0 13) 6 47 81 85
E-Mail: pilz_msolis@infosel.net.mx

NL
Pilz Nederland
Postbus 186
4130 ED Vianen
Netherlands
Telephone (03 47) 32 04 77
Telefax (03 47) 32 04 85
E-Mail: info@pilz.nl

P
Pilz Industrieelektronik S.L.
Apartado 2028
2706-909 Colares
Portugal
Telephone (21) 9 28 91 09
Telefax (21) 9 28 91 13
E-Mail: pilz@esoterica.pt

PRC
Pilz China Representative Office
Flat F9/F Huijing Building
134 Siyou Xin Malu
Dongshan District
Guangzhou 510600
China
Telephone (0 20) 87 37 16 18
Telefax (0 20) 87 37 35 55
E-Mail: pilzchn@public.guangzhou.gd.cn

ROK
Pilz Korea Liaison Office
102-1402 Ilsung apt
767, Kyomun-Dong, Kuri-Si
Kyungki-Do 417-715
Korea
Telephone (31) 5 54 12 80
Telefax (31) 5 54 12 80
E-Mail: pilzkr@hotmail.com

S
Pilz Skandinavien KS
Energigatan 10 B
43437 Kungsbacka
Sweden
Telephone (03 00) 1 39 90
Telefax (03 00) 3 07 40
E-Mail: pilz@tripnet.se

SGP
Pilz Industrial Automation Pte Ltd.
61, Kaki Bukit Ave 1, #05-01
Shun Li Industrial Park
Singapore 417943
Singapore
Telephone 8 44 44 40
Telefax 8 44 44 41
E-Mail: sales@pilz.com.sg

USA
Pilz LP
7150 Commerce Boulevard
Canton
Michigan 48187
USA
Telephone (7 34) 3 54-02 72
Telefax (7 34) 3 54-33 55
E-Mail: info@pilzusa.com

...
In many countries we are represented by sales partners. Please refer to our Homepage for further details or contact our headquarters.

www
www.pilz.com

Internet enquiries and orders: www.pilz.com

Pilz GmbH & Co.
Felix-Wankel-Straße 2, 73760 Ostfildern, Germany
Telephone +49 (7 11) 34 09-0, Telefax +49 (7 11) 34 09-1 33

17 940-01/02 Printed in Germany

Displays and Operator Terminals

PX 30 and PX 120
Operating Manual
Item No. 17 940

In the interest of continual technical advancement we reserve the right to amend technical details without prior notice. No responsibility accepted for errors or omissions. We are grateful for any feedback on the contents of this manual.

III, 01/02

Contents

Contents

Appendix 7-1

Notes

Safety Regulations

General Safety Regulations

- Electrical connections must be made by a qualified electrical engineer who is familiar with the operating manual and the valid regulations for safety in the workplace. VDE and local regulations must be observed, in particular with regard to safety.
- It is important to keep within the permitted operating temperature range.
- Do not open the unit, otherwise all warranty becomes void. Units requiring repair must be returned to Pilz.
- The correct function of the unit is guaranteed only for the operating modes specified in the operating manual. Incorrect connections may damage or destroy the unit or machinery.

Unit-Specific Safety Regulations

You must comply with the following safety regulations in order to ensure the correct operation of your unit:

- When selecting where to install your display, please remember to keep as large a distance as possible between the unit and any electromagnetic fields. This is especially important when frequency converters are nearby. We also recommend you use a bulkhead to separate the display from any source of interference.
- Inductive components built into the surrounding area (eg. contactor, relay and solenoid valve coils), must be wired with an RC-network, especially if they are fed from the same source.
- Data and power lines should be installed separately to avoid capacitive and inductive transmission (recommended minimum distance = 10 cm/ 3.94")!
- To avoid the build-up of heat, a distance of 10 cm/3.94" should be maintained all round the unit.

Safety Regulations

- Interference voltages accessing the unit via supply and signal lines, and electrostatic voltages passed on through contact, are diverted to the earth point (spade terminal on the rear of the unit).
  Connect the earth point (⏚) to the earth conductor PE using as short a copper conductor as possible (≥ 6 mm² csa).

- The earth conductor should be connected to the terminal marked ⏚ on the supply voltage connector (1.5 mm² diameter).

- Before applying voltage always check that the voltage information given on the unit matches your supply.

- Shielded, twisted pair cable must be used for connections to the interfaces.

- Always use metallic or metallised synthetic connectors with the connection cable.

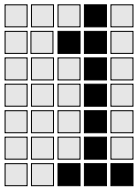
- We recommend that the shielded connection on the network cable between two subscribers is connected on both sides. This will require sufficient potential equalisation between the earth conductor potentials at each subscriber. The most effective method of screening is to connect the braided screening to a potential equalisation rail over a wide surface area. Both the unscreened wires at the cable ends and the shielded connections should be kept as short as possible.
  Please note: On the display system, the cable screening is connected directly (galvanically) to the housing.

  If the potential equalisation is insufficient, the screening on the connection and/or programming cable may only be connected on one side of the connector housing.
  Please note: The screening on the programming cable supplied by Pilz is connected on one side (PX-side) of the connector housing.

- Please ensure that the mounting screws on the connector are firmly attached.

- The power supply for the text display must be designed for a peak switching current of 15 x I_N.
- Connections / units must not be removed or modifed during operation. Always ensure that the supply is switched off.
- Do not change the text memory while power is applied to the unit.

Before installation you should also check the safety requirements laid down by the PLC manufacturer.



Notes

Operation

The PX 30 and PX 120 can be driven in parallel through a PLC's I/O-level or in serial through the V24 interface of a PLC or PC.

The PX 30 has 16 inputs and the PX 120 has 24 inputs available for parallel drive, plus one output.
The PX 120 can be configured to use just 16 inputs (instead of 24) for communicating data and commands. The configuration of "16-input mode" is explained in the chapter entitled "Configuration" from page 6-1 onwards.

Both units (PX 30 and PX 120) can be operated as either a:

- Text Display or
- Text Monitor.

You can select the mode (display or monitor mode) in which the display unit will operate via input E15 on the PX 30 and input E23 on the PX 120:

- E15 or E23 high = monitor mode
- E15 or E23 low = display mode

In display mode, the PLC selects the text stored in the memory through its text number, and variable values sent from the PLC are inserted in the displayed text.

In monitor mode, texts are sent from the PLC as characters from the IBM character set (extended ASCII character set). Control commands such as cursor positioning and the allocation of attributes, etc. must be sent as part of the message.

The features of the two different display modes are described in the table below. The two right-hand columns tell you where you can find additional information on the individual features.

Operating in **Display Mode**

Feature	Refer to Chapter / Section	From Page
Both parallel and serial drive is possible:		
- parallel drive via 24 inputs (PX120)	Parallel drive via a PLC's I/O level - display mode: Text display with 24 inputs	3-5
- parallel drive via 16 inputs (PX 30 or PX 120 in 16-input mode)	Text display with 16 inputs	3-12
- serial drive	Serial drive display mode	3-22

Operating in **Display Mode** (Ctd.)

Feature	Refer to Chapter / Section	From Page
Texts stored in the text memory are selected and displayed	(Text in general:) Text in display mode	4-1
Variables (eg. true and set values) can be inserted into the text. These variables are selected via their number and can be displayed with certain attributes (eg. flashing, right justification). Control characters and text masks edited in the text enable you to allocate attributes and position the cursor.	(Variables in general:) Variables in display mode	5-1
Additional display functions (eg. scrolling and segment test) are available via control commands from the PLC.	Driving the Display	3-1
Configuration instructions stored in the text memory enable the display to adapt to certain user requirements (eg. scroll function, coding and the number of inputs used).	Configuration	6-1
Several displays can be networked through a V24 - RS 485 interface adapter.	Appendix / Networking Capabilities	7-6

Operating in **Monitor Mode**

Feature	Refer to Chapter / Section	From Page
Both parallel and serial drive is possible:		
- parallel drive via 24 inputs (PX 120) - parallel drive via 16 inputs (PX 30 or PX 120 in 16-input mode	Parallel drive via a PLC's I/O level - monitor mode	3-20
- serial drive	Serial drive monitor mode	3-31
PLC sends texts as IBM/ASCII characters. These are displayed at the cursor position.		
Additional display functions (eg. scrolling and segment test) are available via control commands from the PLC.		

Features

The following features have been incorporated into the units:

- IBM character set
- Flash-EPROM as text memory, programmable within the PX-unit.
- Either parallel (via I/O-level) or serial (via V24) drive
- Optimum speed
- Shallow modular depth
- Automatic brightness control on the display (dimmer function)
- Extended scrolling options
- Networkable
- Status display

The main difference between the PX 30 and PX 120 is their display format and the number of inputs they have available:

Unit	Display Format	Number of PX Inputs
PX 30	2 lines of 20 characters; 5 mm character height; 5 x 7 dot matrix	16
PX 120	2 lines of 40 characters; 9 mm character height; 5 x 12 dot matrix	24

Data transmission varies depending on the number of inputs. These differences are explained in Chapter 2.

Flash-EPROM Memory

Flash-EPROMs are used as text memory. These are currently the quickest memory chips available for text displays. Just like EEPROM-chips, they are programmed and cleared electrically. Flash-EPROMs can be programmed directly inside the PX-unit, combining the data security of an EPROM with the flexibility of a RAM. The ability of the EEPROM to clear byte by byte is not availabe on the Flash-EPROM, but this facility is not required on text displays.

The simplest way to create text is to use an IBM-compatible PC with ADIT DOS (version 4.0 and above). This software has been specially designed for organising data transmission on these PX-units. Texts, attributes for text and variables (eg. flashing representation) and text masks for variables are all edited through dialogue.

The Flash-EPROM is programmed directly inside the PX-unit. When the text display is ready for operation, (ie. 24 V is supplied), just connect it up to a PC through its serial interface and the text memory can be programmed quite simply using the ADIT DOS Text Editor (under MS DOS).

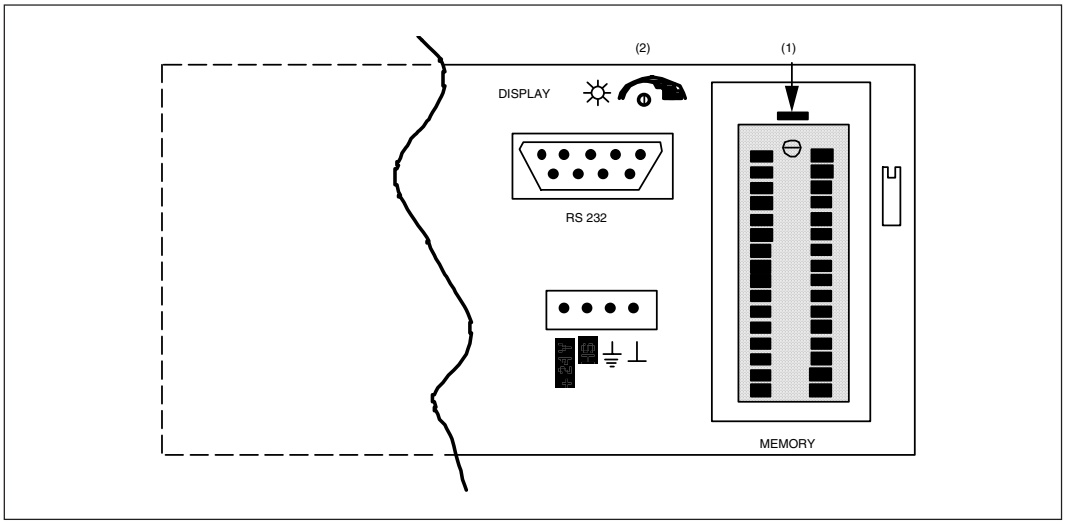
The programming of the Flash-EPROM in the PX-unit via the serial interface (V24) is not described in this manual. For further details please refer to the ADIT DOS Manual.

System Description PX 30 and PX 120

Dimmer Function

A sensor judges the brightness at the front of the unit and adapts the brightness of the display accordingly.

This dimmer function is activated by setting a jumper (1) on the rear of the unit (below the Flash-EPROM).



The jumper is not inserted ex-works, so the dimmer is switched off. In this case the display will operate at a constant maximum brightness.

When the dimmer is activated (jumper inserted), optimum standard brightness is achieved as follows:
Adjust the dimmer setting (2) with a screwdriver until maximum brightness is achieved under typical light conditions.
Start from the right stopper and turn anti-clockwise until the display begins to darken.
Then carefully turn it clockwise again until it lightens.

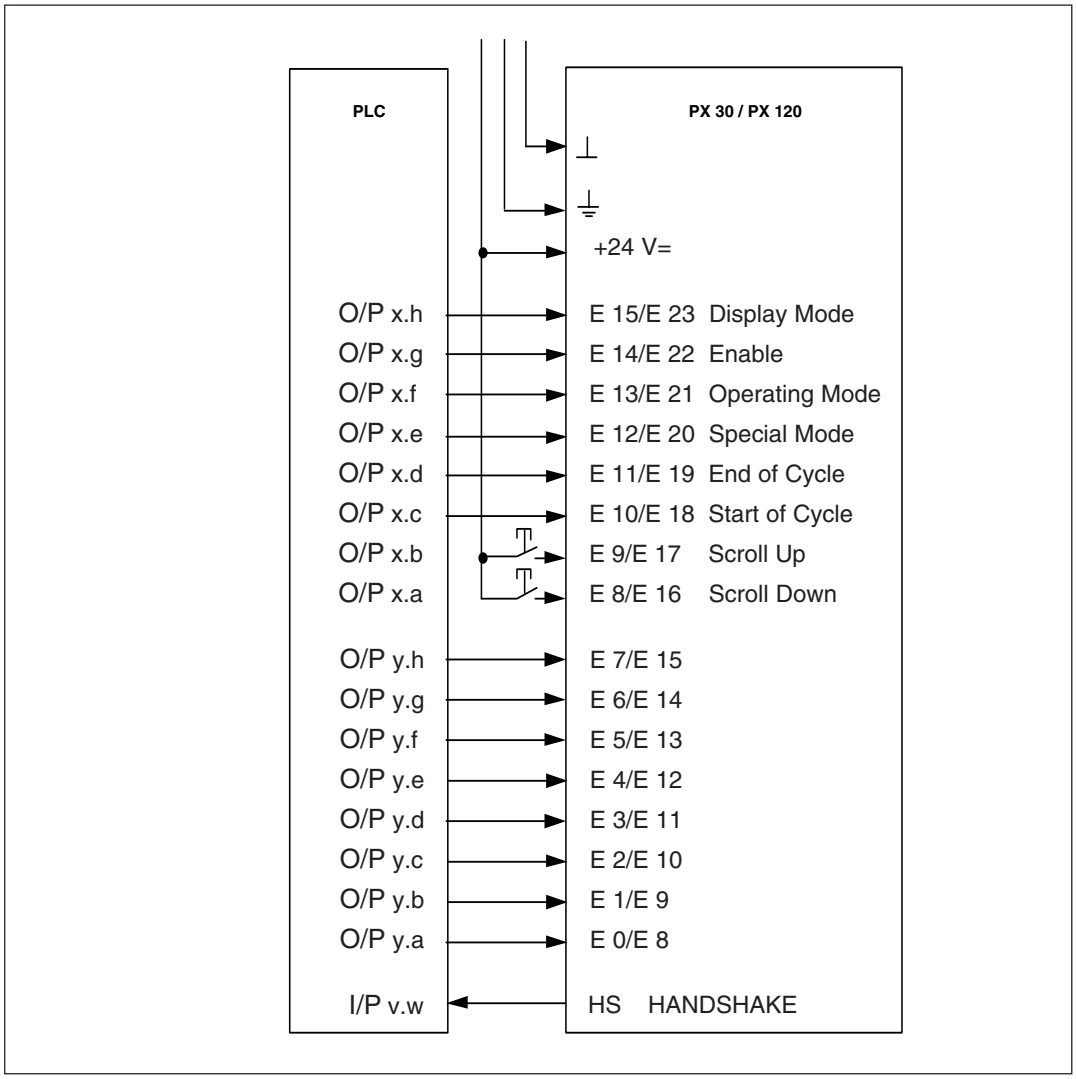
Hardware

Both units have:

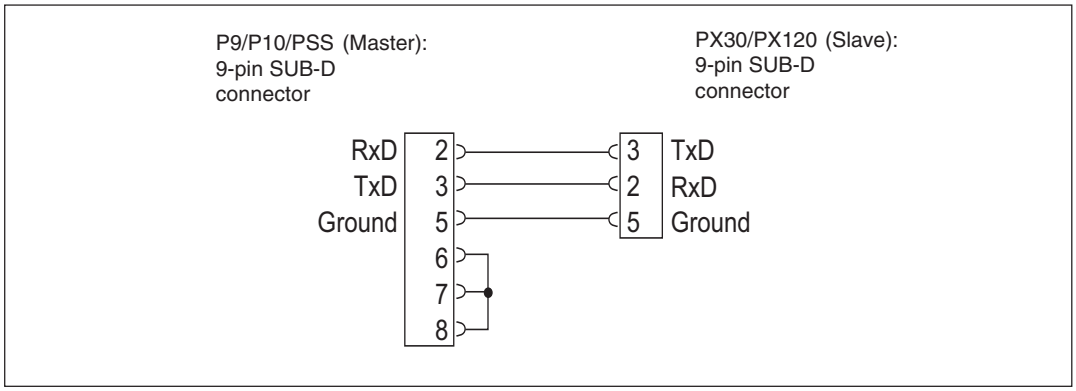
- Fluorescent Display; 2 lines
- 24 V inputs (16 inputs on the PX 30, 24 inputs on the PX 120)
- One 24 V output (handshake output "HS")
- Serial Interface (V24, 9-pin)
- Automatic brightness control (dimmer function)
- 32 KByte Flash-EPROM cartridge; 64 KByte available as an option
- Ability to program the Flash-EPROM inside the unit.

PLC Connections

Parallel drive via the I/O-level:



Serial drive via the V24 Interface:



Notes

Driving the Display

Handshake Function

Important: The handshake function can only be applied when driving the display through a PLC's I/O-level in parallel mode!

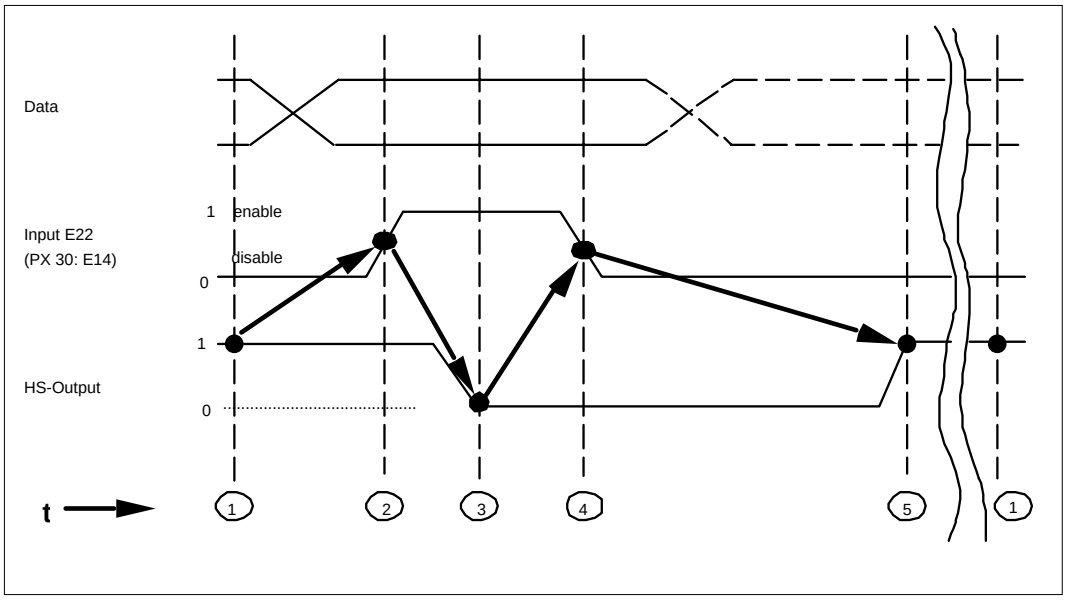
When the display is driven in parallel (in display and monitor mode), data communication occurs in the handshake algorithm, which serves the handshake output HS. Its status signalizes:

- High-level: Display is ready to receive
- Low-level: Display is not ready to receive

Data and commands received by the PX are validated with an enable signal (positive-going pulse edge at E22 on the PX 120 and E14 on the PX 30).
Exception: Scroll inputs E16/E17 (PX 120) and E8/E9 (PX 30) operate without an enable signal. They merely require a positive-going pulse edge at the relevant input.

The diagram overleaf illustrates the handshake function.

Driving the Display



(1) HS-output = 1: New data can be supplied.

(2) Input E22 (PX 30: E14) may be set from 0 to 1 no earlier than the time at which the data arrives (enable signal).

(3) As a result of the positive-going pulse edge at E22 (PX 30: E14), new data is read in from the PX-unit and the HS-output is automatically set at 0.

(4) Only then is E22 (PX 30: E14) returned to 0. Only at this point can new data be supplied.

(5) As a result of the negative-going pulse edge at E22 (PX 30: E14), the HS-output automatically returns to 1, but only once the data is processed by the PX-unit. A Logic 1 signal at the HS-output signals that the unit is ready to receive new data (see (1)).

Operating Modes in Display Mode

The following operating modes are supported when the PX is in display mode, with both parallel and serial drive:

- Text mode
- Variable mode
- Special mode

In text mode, texts are selected and displayed by means of their text number, which can be either binary or BCD-coded.
Variable mode is used for the selection, evaluation and insertion of variables. Variable transmission may be binary, BCD or ASCII-coded.

For further details of codes and their features, please refer below to the section entitled "Coding".

In special mode, the following special functions are supported:

- Segment test
- Output of version number
- Automatic display of the text memory contents
- Clear display
- Status display

Parallel Drive through a PLC's I/O-level

Display Mode

Coding

For parallel drive of the PX 30 and PX 120 in display mode, two base settings can be configured for data coding:

- Binary-coding (default)
- BCD-coding

The required coding method (binary or BCD) is set in the $M-configuration instruction (see section entitled "Configuration" on page 6-1) and is then used for communicating text numbers (text selection in text mode).

If no $M-configuration instruction is present, both the PX 30 and PX 120 will operate in binary code in display mode (default setting).

In variable mode when binary coding has been configured, or the default setting is un-changed, variable values can be transmitted in three ways:

- Binary-coded
- BCD-coded or
- ASCll-coded

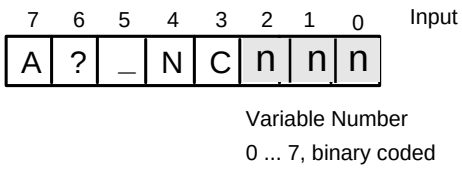
If BCD-coding has been configured, variable values can be transmitted in two ways:

- BCD-coded or
- ASCll-coded

The variable code is selected in the 1st cycle of each new variable transmission.

The diagram overleaf shows the function and layout of inputs E0 - E7 in the 1st cycle of each variable transmission (ID-Byte).

Driving the Display



A = 1	Display ASCll-coded variables
A = 0	Display binary or BCD-coded variables
? = 1	"?" flashes alternately with the lowest value digit of the variable value
? = 0	"?" not inserted
_ = 1	"_" flashes alternately with the lowest value digit of the variable value
_ = 0	"_" not inserted
N = 1	A negative prefix "-" is placed before the variable value
N = 0	Display without negative prefix
C = 1	BCD-coding
C = 0	Binary-coding

If E3 = 0, binary coding is selected, if E3 = 1, BCD-coding is selected.

If BCD-coding is selected, the maximum number of BCD-digits per cycle will be transmitted in parallel operation. The maximum number is 4 BCD-digits in 24-input mode and 2 BCD-digits in 16-input mode, provided the $M-configuration instruction has not been used to reduce the number of digits which can be transmitted in parallel operation in BCD-mode (see page 6-1).

Important: Input E3 has no function when BCD-coding is configured.

Please note:

- The configuration for the number of inputs used on the PX 120 (24 or 16-input-mode) is described in the section entitled "Configuration" from page 6-1 onwards.
- If input E4 is latched, a negative prefix can be placed before a BCD-digit. Binary mode operates with signed integers.
- The conversion of a positive binary integer into a negative binary integer of the same value (taking into account the prefix coding in the highest value bit) is carried out as follows:

1. Positive binary integer
2. Complementing (negation) of binary value
3. Add "1"

Example: "-21" as a signed 8-bit binary figure

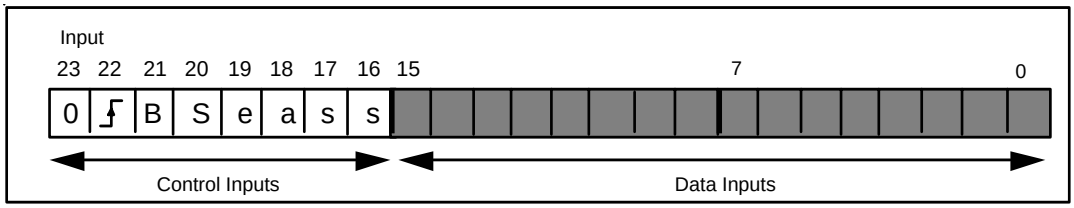
1. "+21": 00010101
2. 11101010
3. + 00000001
4. 11101011 (on PX-unit): "-21"

Important: You will need to be in variable mode to use input E3 to switch between binary and BCD-coding. Even in this case, binary coding will need to have been configured, or the $M-configuration instruction will need to have been omitted!

Text Display with 24 Inputs (PX 120)

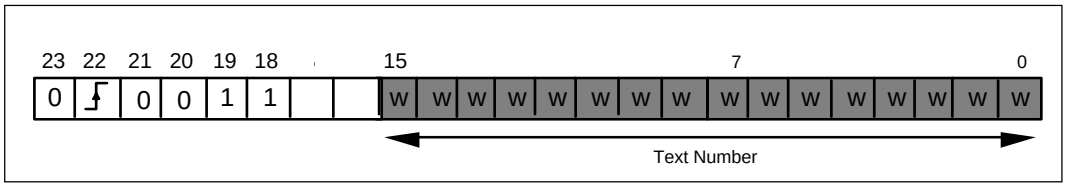
Note: The PX 120 can be configured to use just 16 inputs (see page 6-1), making it fully compatible with the PX 30. The function and layout of inputs in 16-input mode is described from page 3-12 onwards.

Function and Layout of Inputs (general)



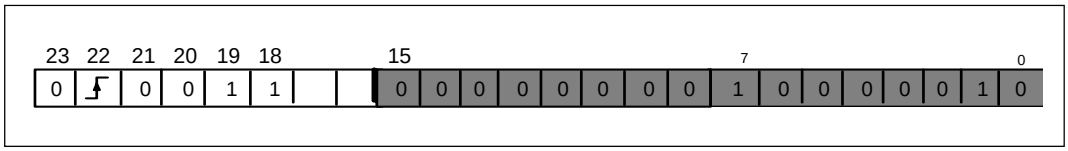
E23 = 0	Display Mode
⎍	Enable signal (positive-going pulse edge)
B/S	Operating Mode/Special Mode: 0/0 = Text Mode; 0/1: Special Mode; 1/0 = Variable Mode; (1/1 = reserved)
e	End of data transmission: 1 = last cycle
a	Start of data transmission: 1 = first cycle
s	Scroll inputs (E17 up; E16 down)

Text Mode
Binary Coding

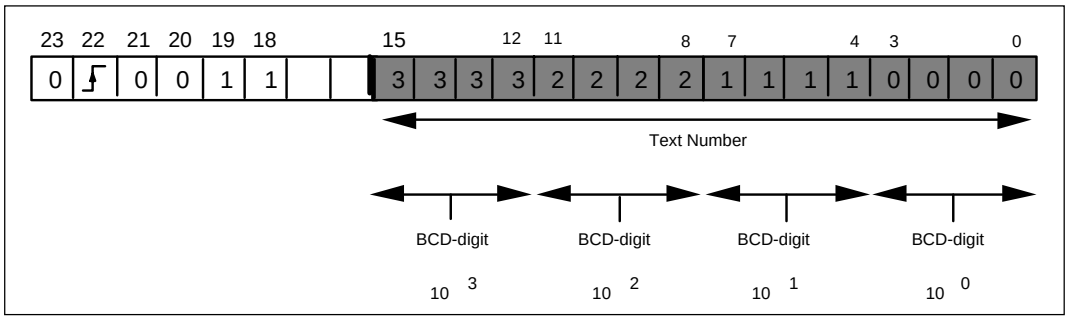


The text number is interpreted as a signed 16-bit integer. Text numbers from 0 ... 9999 are accepted.

Example: Calling up the message with text number 130

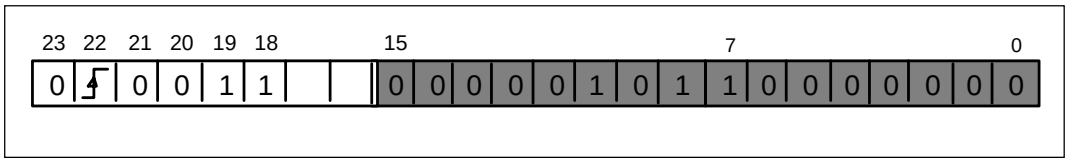


Text Mode
BCD-Coding

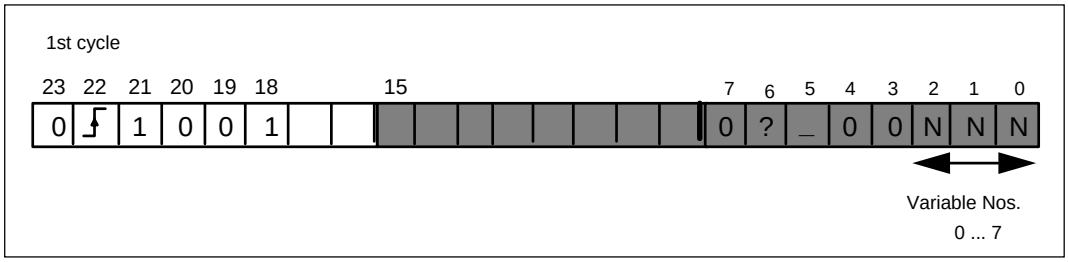


Text numbers from 0 ... 9 999 are accepted.

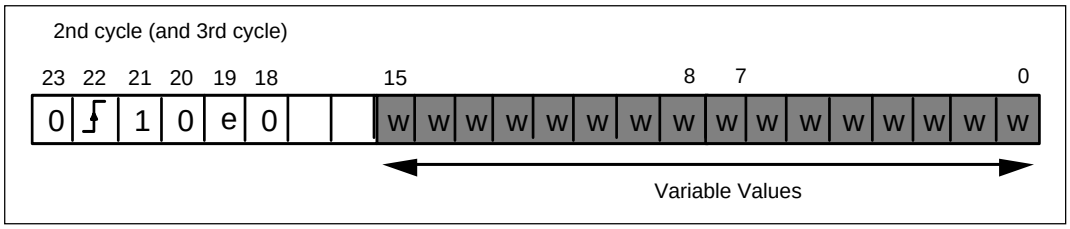
Example: Calling up the message with text number 580



Variable Mode
Binary-Coded Transmission



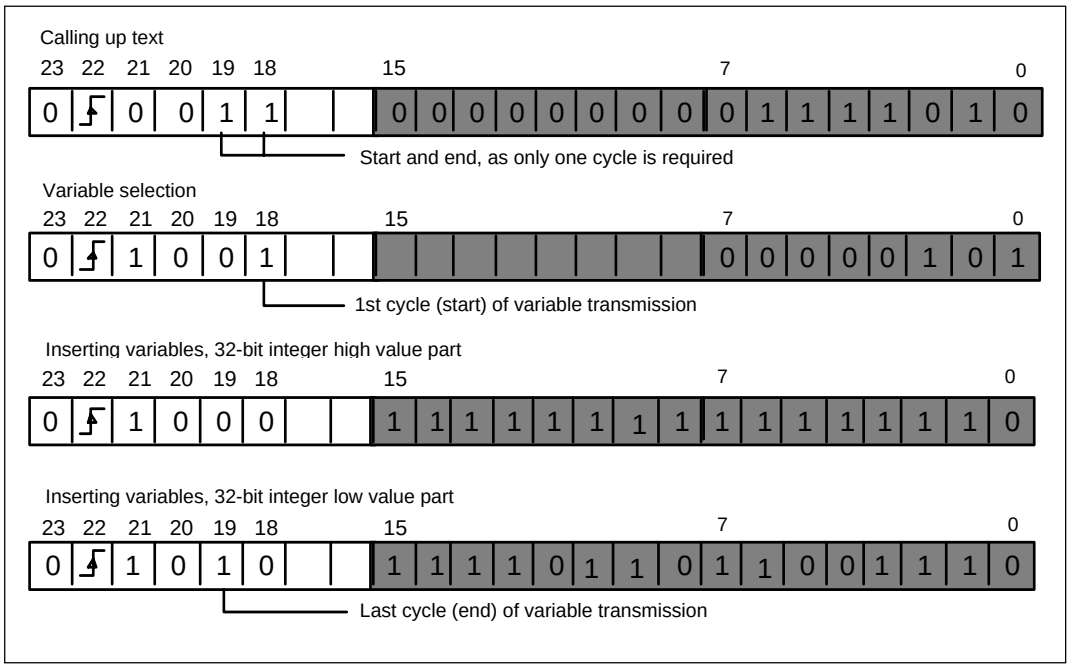
? = 1	Inserts "?"; flashes alternately with the lowest value digit of the variable value
? = 0	"?" not inserted
_ = 1	Inserts "_"; flashes alternately with the lowest value digit of the variable value
_ = 0	"_" not inserted
If ? = 1 **and** _ = 1, "?" will be inserted. "?" or "_" can still be flashed at the point of the lowest value digit even when the variable value is not transmitted (only in the 1st cycle, when E18 and E19 = 1).	



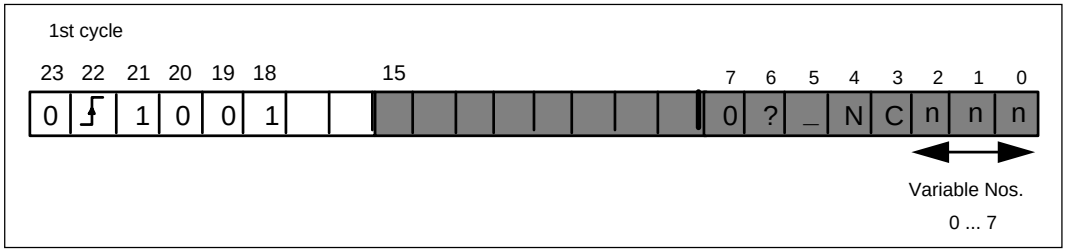
Second cycle only: signed 16-bit integer (value range: -32 768 ... +32 767)
Third cycle: additional 16-bit as low value part of a signed 32-bit integer
(value range: -2 147 483 648 ... +2 147 483 647)

The last cycle is indicated when e = 1.

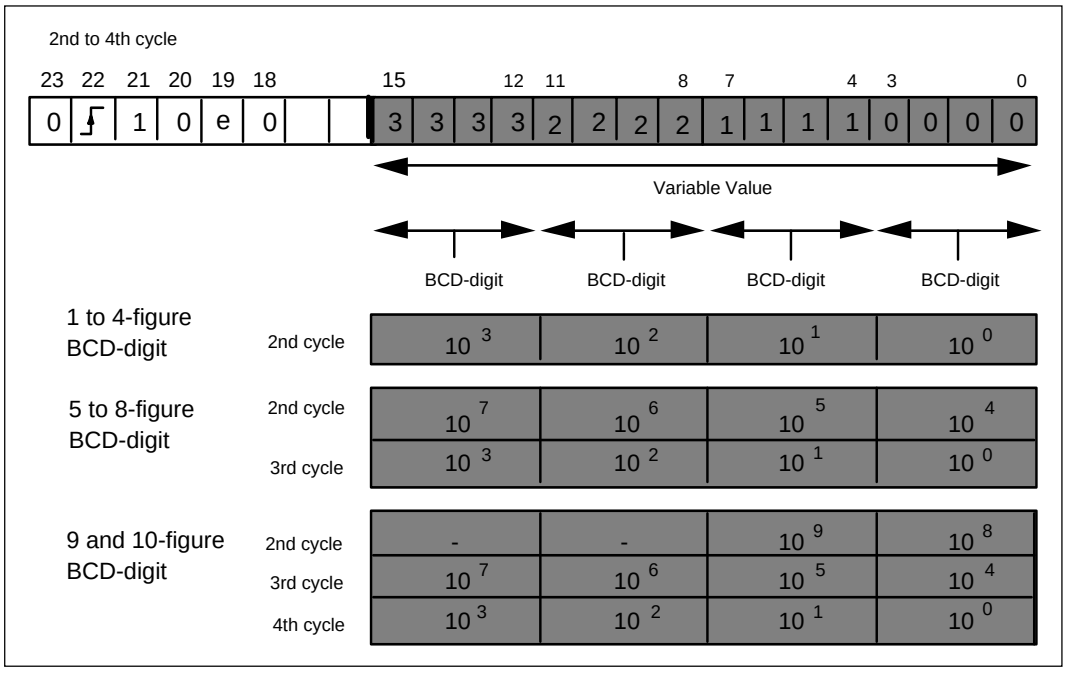
Example: Inserting the value -67 890 for variable 5 in the message with text number 122



Variable Mode
BCD-Coded Transmission

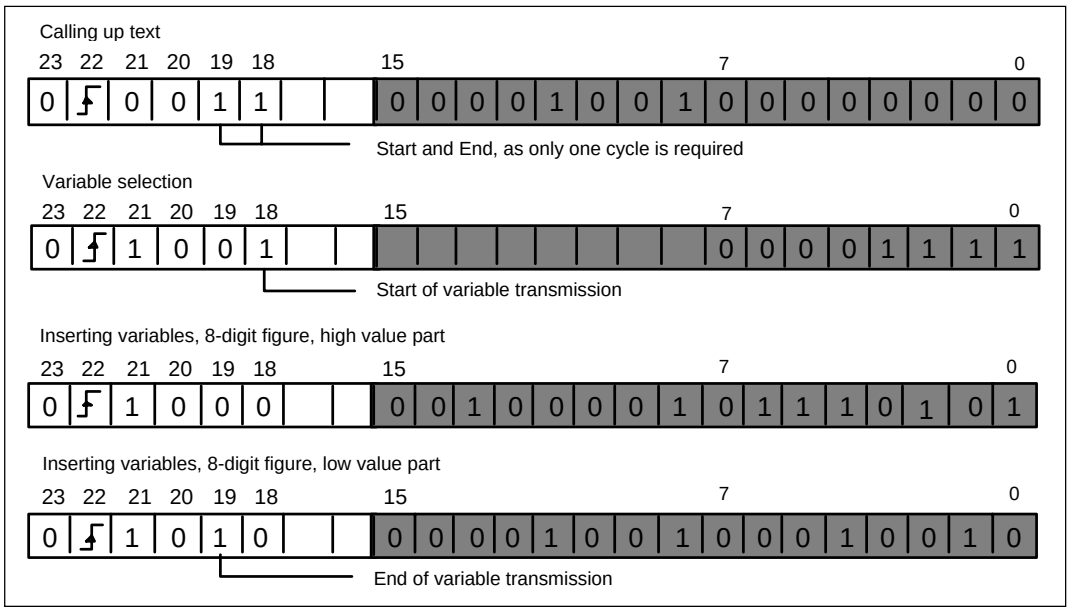


? = 1	Inserts "?"; flashes alternately with the lowest value digit of the variable value
? = 0	"?" not inserted
_ = 1	Inserts "_"; flashes alternately with the lowest value digit of the variable value
_ = 0	"_" not inserted
If ? = 1 **and** _ = 1, "?" will be inserted. "?" or "_" can still be flashed at the point of the lowest value digit even when the variable value is not transmitted (only in the 1st cycle, when E18 and E19 = 1).	
N = 1	A negative prefix (-) is placed before the variable value
N = 0	Display without negative prefix
C = 1	Display configured for binary-coding
C = 0/1	Display configured for BCD-coding

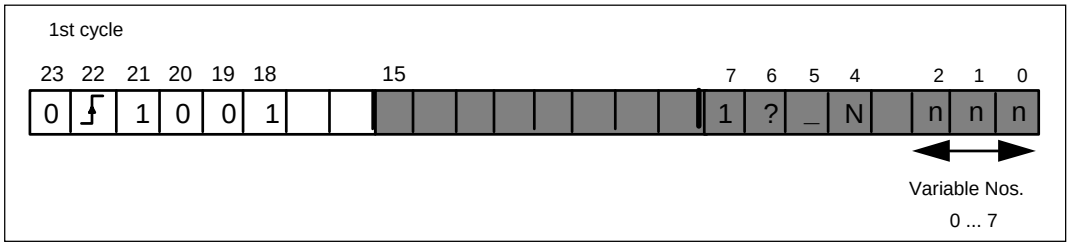


From the 2nd cycle onwards, the variable value is transmitted from the higher value to the lower value part. In the last cycle (in which the lowest value BCD-digits are sent), a Logic 1 signal at E19 (e) signifies the end of variable transmission. When more than 8 BCD-digits are to be transmitted (10 BCD-digits = maximum variable length), variable values will need to be inserted in three cycles. In the 1st of these cycles, the 2 highest value BCD-digits will be ignored (see diagram on page 3-8).

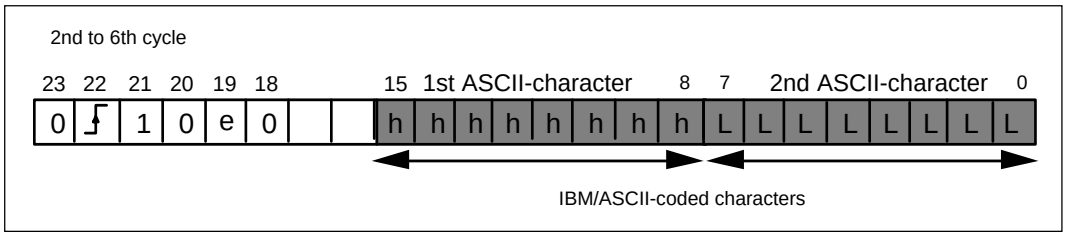
Example: Inserting the value 21 750 912 for variable 7 in the message with text number 900.



Variable Mode
ASCII-Coded Transmission



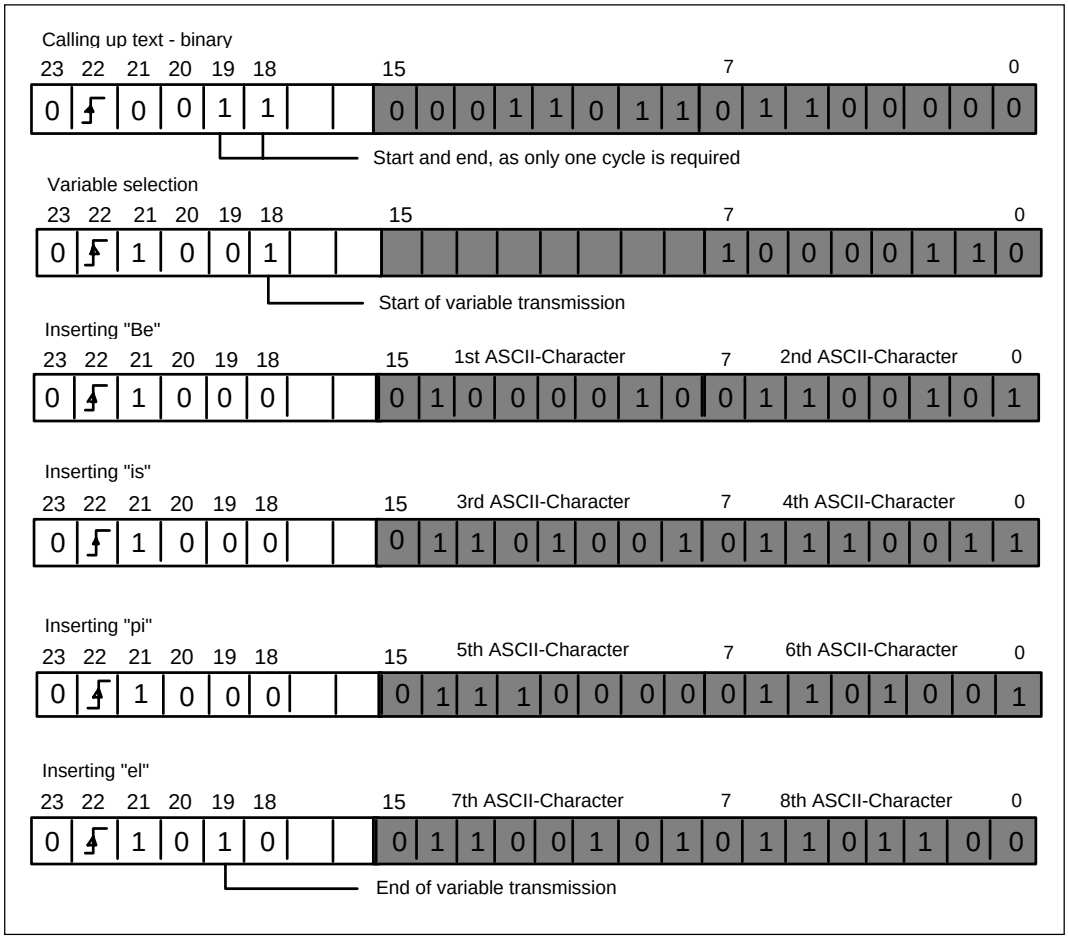
? = 1	Inserts "?"; flashes alternately with the lowest value digit of the variable value
? = 0	"?" not inserted
_ = 1	Inserts "_"; flashes alternately with the lowest value digit of the variable value
_ = 0	"_" not inserted
If ? = 1 **and** _ = 1, "?" will be inserted. "?" or "_" can still be flashed at the point of the lowest value digit even when the variable value is not transmitted (only in the 1st cycle, when E18 and E19 = 1).	
N = 1	A negative prefix (-) is placed before the variable value
N = 0	Display without negative prefix



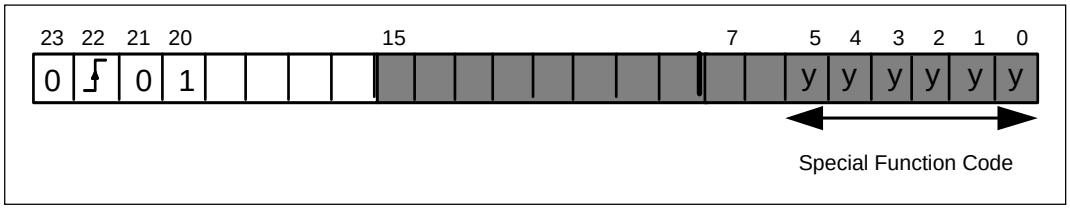
The first ASCll-character is displayed at high value position, the second ASCll-character at low value position (to the right of the first character). If the first ASCll-character is 00H, only the second ASCll-character is transmitted.

The last cycle is indicated when e = 1. A maximum of 10 ASCll-characters per variable is accepted. The last ASCll-characters transmitted are displayed on the right hand side of the preceding ASCll-characters.

Example: Inserting the word "Beispiel" (example) for variable 6 in the message with text number 7008.



Special Mode



Special Function Code		Function
binary	hexadecimal	
000000	00	Not occupied
000001	01	Segment test on
000010	02	Segment test off
000011	03	Output version number
000100	04	Auto-display of text memory contents - on
000101	05	Auto-display of text memory contents - off
000110	06	Clear display
000111	07	Status indicator on

- **Segment Test**:
  All pixels on the display are driven; those that are intact will flash! When the segment test is switched off the last text to be displayed will reappear. System messages displayed previously will not reappear!

  Please note: To maintain equal brightness on all display segments, we recommend that you run the segment test every week for approximately 3 hours.

- **Output Version Number**:
  The version number will be displayed in the following format: "PX 30, Version x.xx" or "PX 120, Version x.xx". This is the default message on power-up.

- **Auto display of text memory contents**:
  When this is switched on, all records from 1 ... 9999 are automatically displayed one after the other. Available records are displayed with their text; for records that are not available, the text number will appear in the format: "xxxx" (xxxx ... corresponding text number). The unit will automatically switch off after record number 9 999 is displayed.

- **Clear display**:
  The display and all variable positions are cleared.

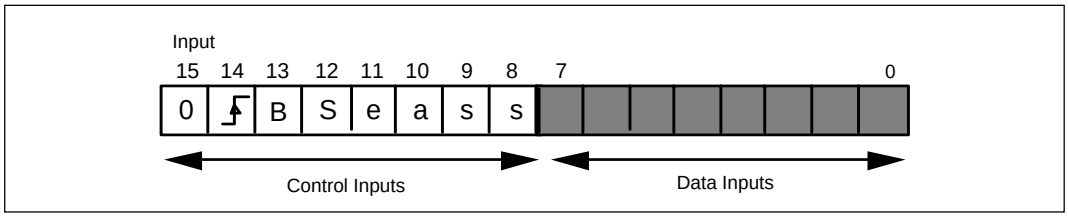
- **Status display**:
  The current status of data inputs (PX 30: E7 ... E0; PX 120: E15 ... E0) are shown on the display as "0" and "1". The lowest value input is shown on the right hand side. The status display can be switched off at control inputs E23 ... E16, using the code $5C_H$ (0101 1100 binary).

The function and layout of inputs in special mode is the same for binary as for BCD-mode.

Text Display with 16 Inputs (PX 30 or PX 120 in 16-Input Mode)

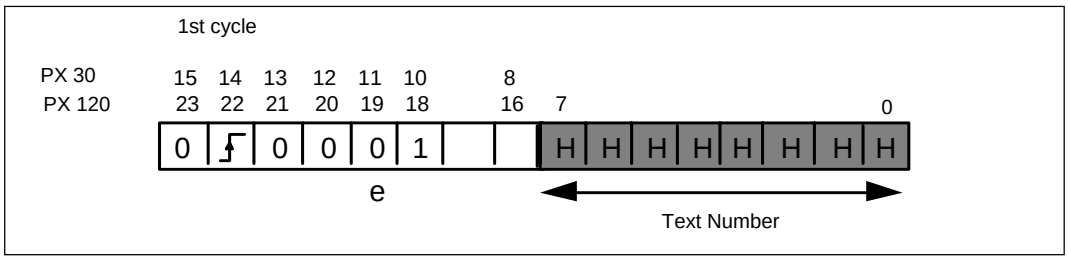
Note: Data Inputs E8 ... E15 are not used when the PX 120 is in 16-input mode. Control inputs are E16 ... E23. Data inputs are E0 ... E7.

Function and Layout of Inputs (general)

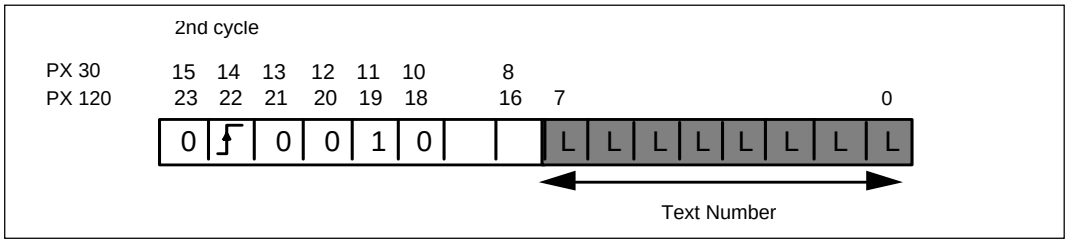


E15 = 0	Display Mode
⨍	Enable signal (positive-going pulse edge)
B/S	Operating Mode/Special Mode: 0/0 = Text Mode; 0/1: Special Mode; 1/0 = Variable Mode; (1/1 = reserved)
e	End of data transmission: 1 = last cycle
a	Start of data transmission: 1 = first cycle
s	Scroll inputs (E9 up; E8 down)

Text Mode
Binary-Coding

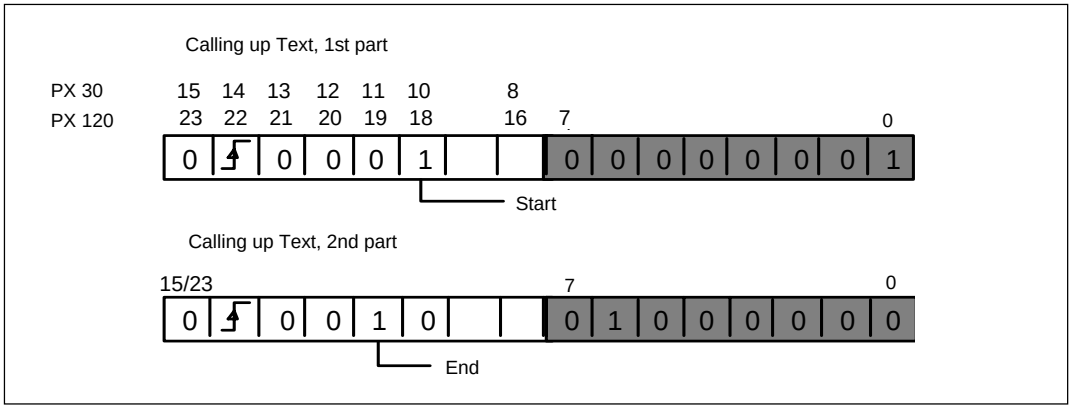


In the first cycle, the high value byte of a signed 16-bit integer is transmitted. If only one cycle is required (for text numbers 0 ... 127), Input e should be a Logic 1 Signal in the first cycle.

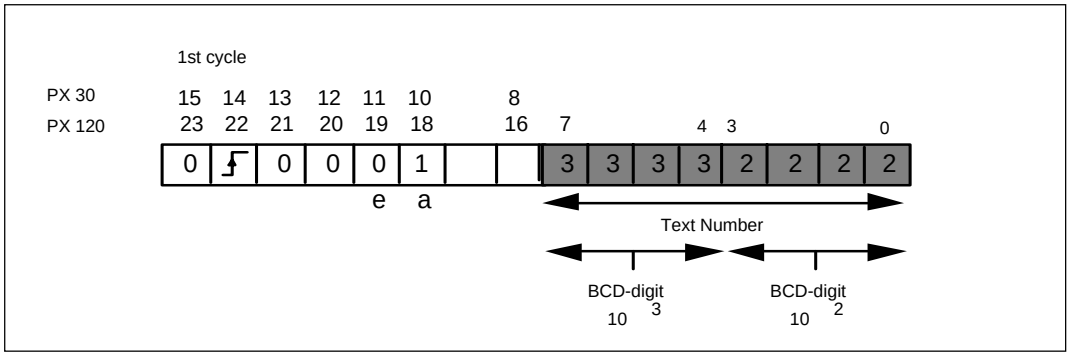


Second cycle: low value byte of a signed 16-bit integer. Text numbers from 0 ... 9999 are accepted.

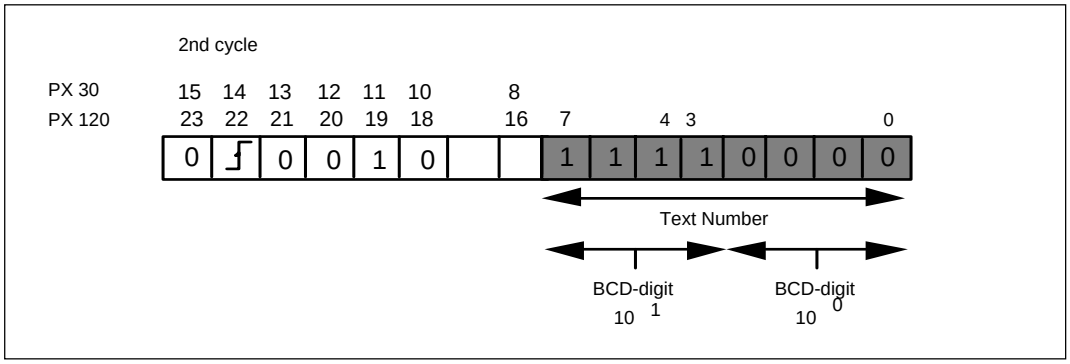
Example: Calling up the message with text number 320.



Text Mode
BCD-Coding

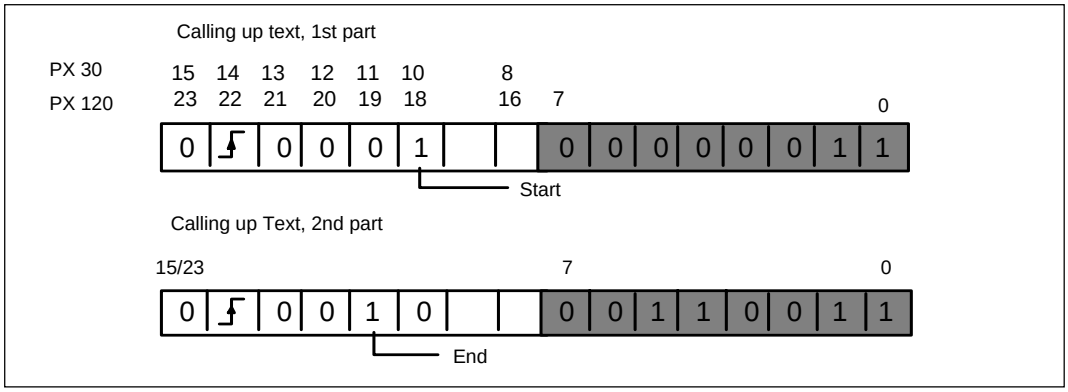


If inputs a and e = 1 in the first cycle, only the BCD digits 10^1 and 10^0 will be transmitted.

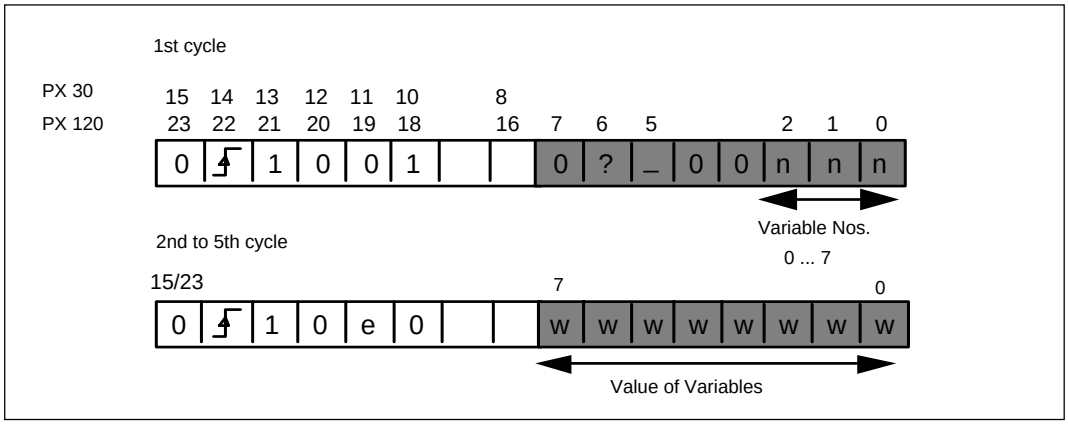


Text numbers from 0 ... 9999 are accepted.

Example: Calling up the message with text number 333



Variable Mode
Binary-Coded Transmission



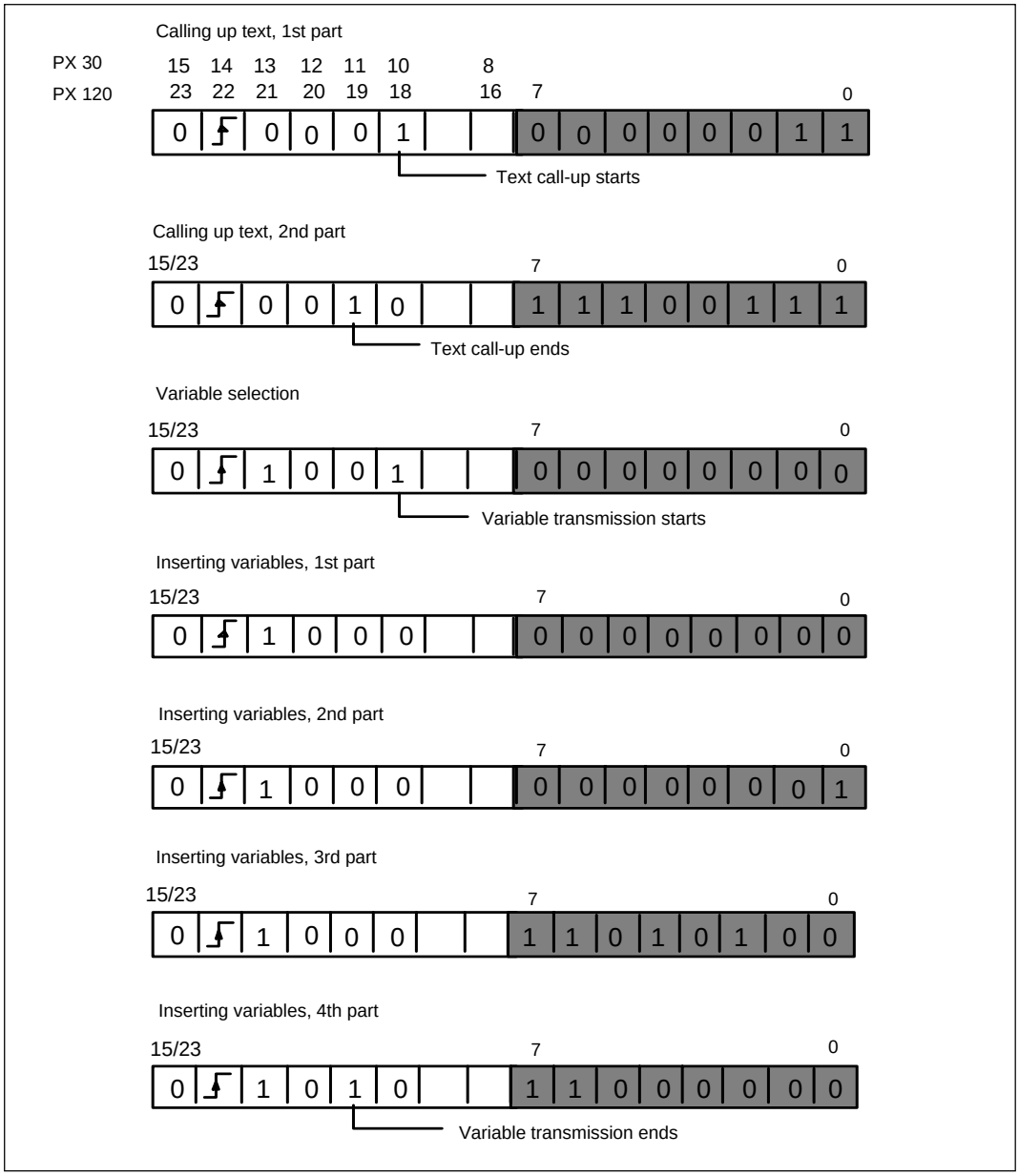
? = 1	Inserts "?"; flashes alternately with the lowest value digit of the variable value
? = 0	"?" not inserted
_ = 1	Inserts "_"; flashes alternately with the lowest value digit of the variable value
_ = 0	"_" not inserted
If ? = 1 **and** _ = 1, "?" will be inserted. "?" or "_" can still be flashed at the point of the lowest value digit even when the variable value is not transmitted (only in the 1st cycle, when E18 and E19 = 1)	

e = 1 in the third cycle: signed 16-bit integer
e = 1 in the fifth cycle: signed 32-bit integer

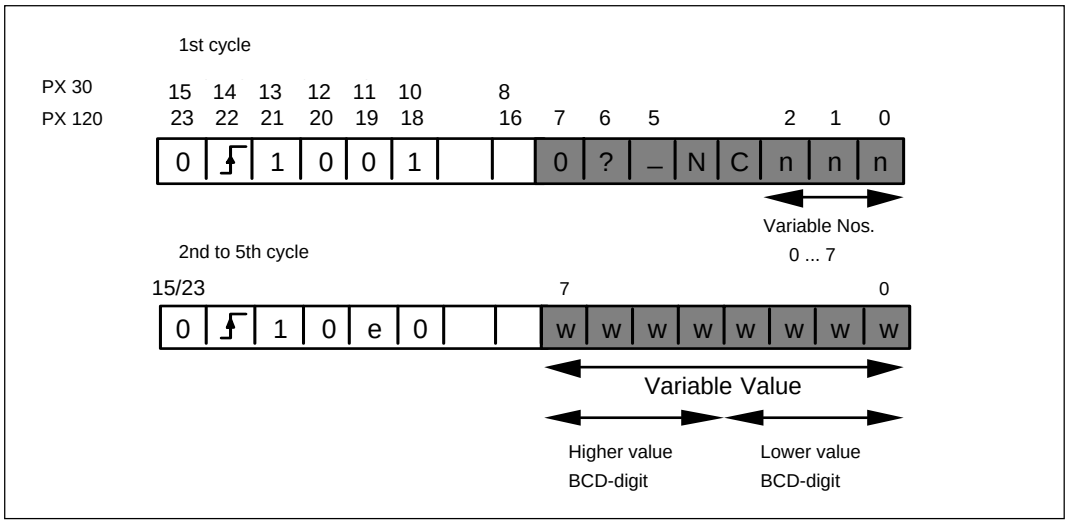
The last cycle is indicated when e = 1.

Transmission occurs from the highest value to the lowest value 8-bit digit of the figure.

Example: Inserting the value 120 000 for variable 0 into text 999



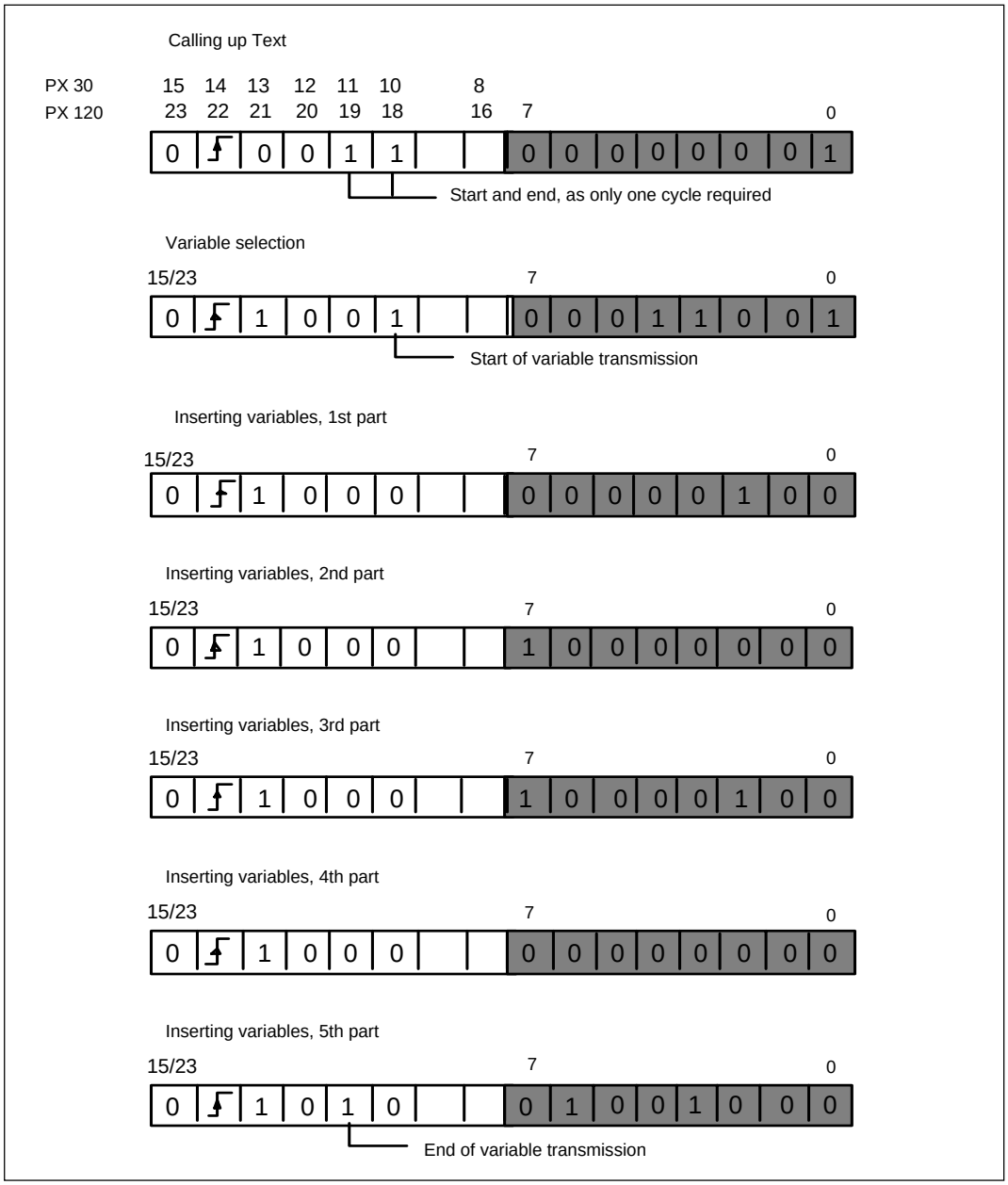
Variable Mode
BCD-Coded Transmission



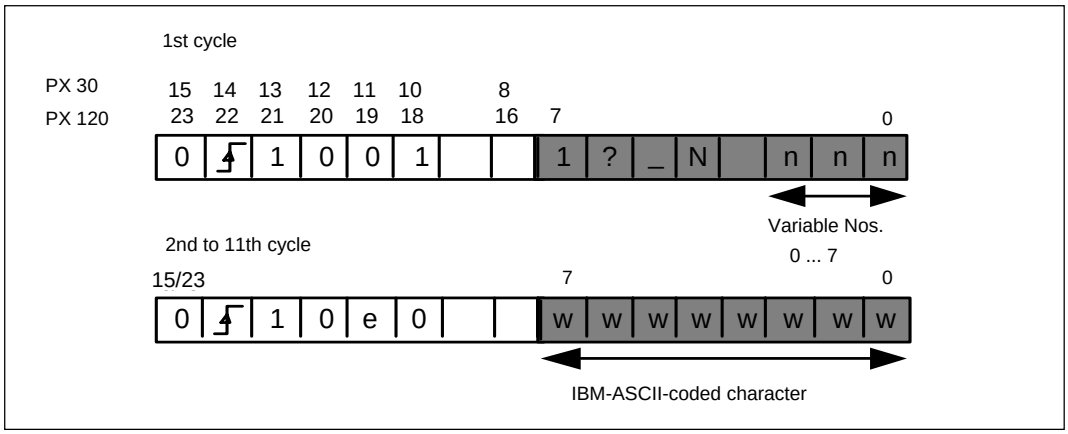
? = 1	Inserts "?"; flashes alternately with the lowest value digit of the variable value
? = 0	"?" not inserted
_ = 1	Inserts "_"; flashes alternately with the lowest value digit of the variable value
_ = 0	"_" not inserted
If ? = 1 **and** _ = 1, "?" will be inserted. "?" or "_" can still be flashed at the point of the lowest value digit even when the variable value is not transmitted (only in the 1st cycle, when E18 and E19 = 1).	
N = 1	A negative prefix (-) is placed before the variable value
N = 0	Display without negative prefix
C = 1	Display configured for binary-coding
C = 0/1	Display configured for BCD-coding

The cycle in which E19 (e) becomes a Logic 1 signal (last cycle of variable transmission) determines the value of the 1st decimal power transmitted. The last two figures transmitted correspond to the 10^1-digit and the 10^0-digit.

Example: Inserting the value -480 840 048 for variable 1 into text 1.



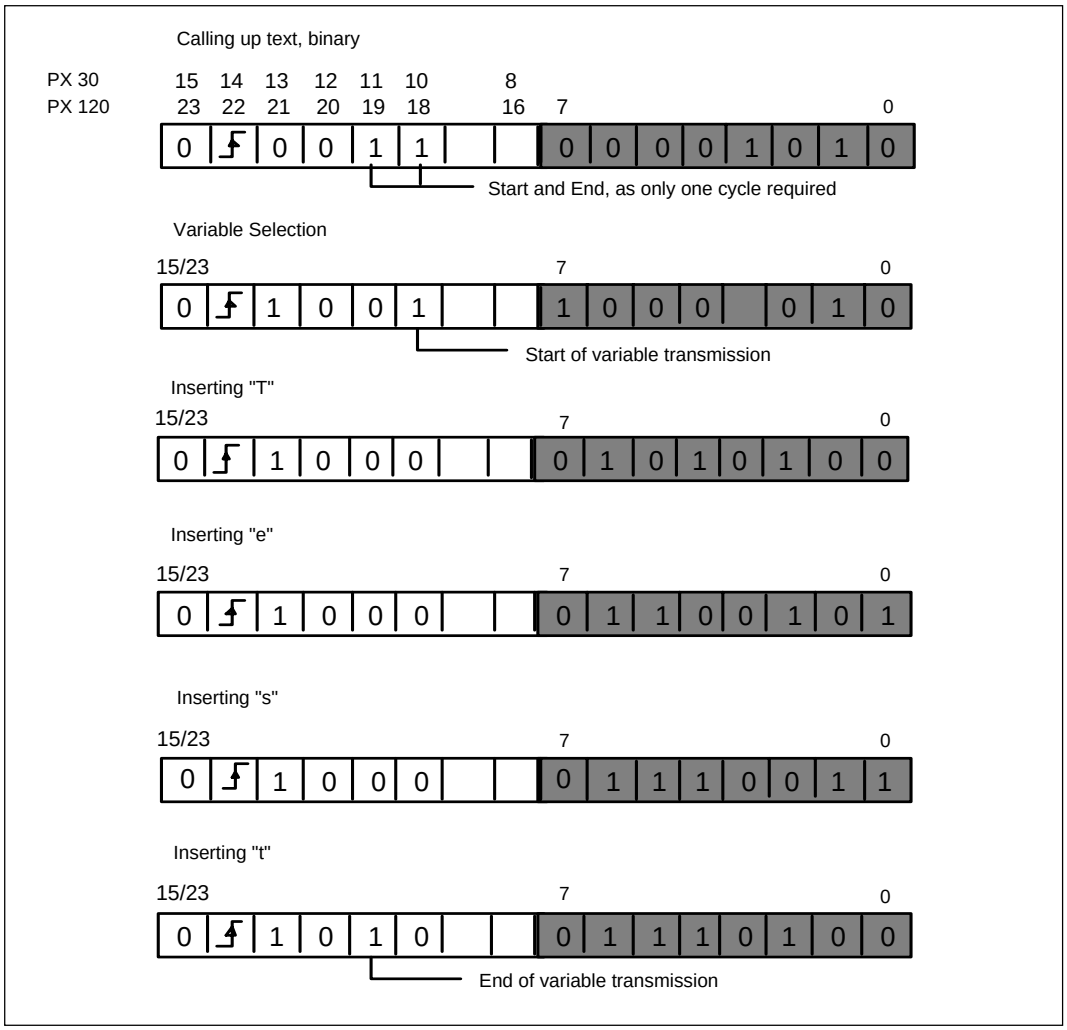
Variable Mode
ASCII-Coded Transmission



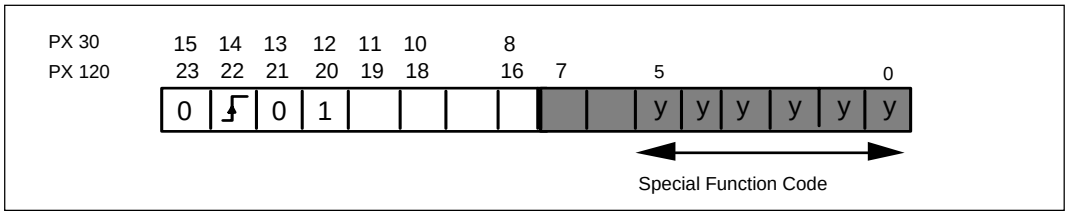
? = 1	Inserts "?"; flashes alternately with the lowest value digit of the variable value
? = 0	"?" not inserted
_ = 1	Inserts "_"; flashes alternately with the lowest value digit of the variable value
_ = 0	"_" not inserted
If ? = 1 **and** _ = 1, "?" will be inserted. "?" or "_" can still be flashed at the point of the lowest value digit even when the variable value is not transmitted (only in the 1st cycle, when E10/18 and E11/19 = 1).	
N = 1	A negative prefix (-) is placed before the variable value
N = 0	Display without negative prefix

The last cycle is indicated when e = 1. A maximum of 10 ASCII-characters per variable is accepted. The last ASCII-character transmitted is displayed on the right hand side of the preceding ASCII-characters.

Example: Inserting "Test" for variable 2 into text 10.



Special Mode



Special Function Code		Function
binary	hexadecimal	
000000	00	Not occupied
000001	01	Segment test on
000010	02	Segment test off
000011	03	Output version number
000100	04	Auto-display of text memory contents - on
000101	05	Auto-display of text memory contents - off
000110	06	Clear display
000111	07	Status display - on

Please refer to page 3-11 for details of the special functions.

The function and layout of inputs in special mode is the same for binary as for BCD-mode.

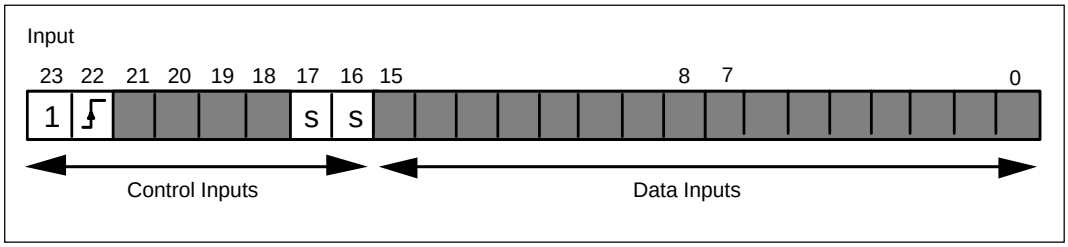
Parallel Drive through a PLC's I/O-Level

Monitor Mode

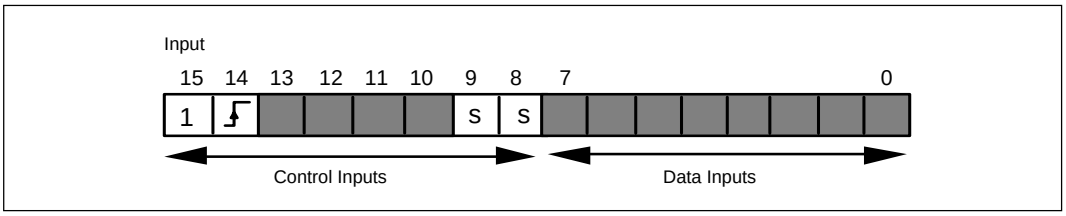
Monitor mode is active when input E23 (PX 120) or E15 (PX 30) is high. Input E22 or E14 (handshake input; enable signal) and the HS output (handshake) are used to drive the handshake algorithm. Inputs E0 ... E15 (PX 120) and E0 ... E7 (PX 30 or PX 120 in 16-input mode) are data inputs.

Function and Layout of Inputs

24 Inputs (PX 120)



16 Inputs (PX 30)



E23/E15 = 1	Monitor mode
⨍	Enable signal (positive-going pulse edge)
E18 ... E21/ E13 ... E10	No function
s	Scroll inputs (E17/E9 up; E16/E8 down)

- Representable IBM characters, displayed at the current cursor position OR
- IBM characters with control functions (control codes for control commands)

are transmitted via the data inputs as 8-bit data.

On the PX 120 (24 inputs) data at inputs E8 ... E15 is processed before data at inputs E0 ... E7.

IBM-characters with control functions can be assigned to the following groups of control commands:

Control Command Group	Example
Cursor Control	Positioning the cursor, cursor visible / invisible
Scroll Control	Scroll up / down
Delete Commands	Delete display
Miscellaneous	Flashing display on / off; determine character set

An overview of the control codes for monitor mode can be found on pages 7-3 and 7-4.

Please note: When the display is in monitor mode, control codes can be used to switch between IBM and Cyrillic character sets (see section entitled "Control Codes" in the Appendix).

Driving the Display

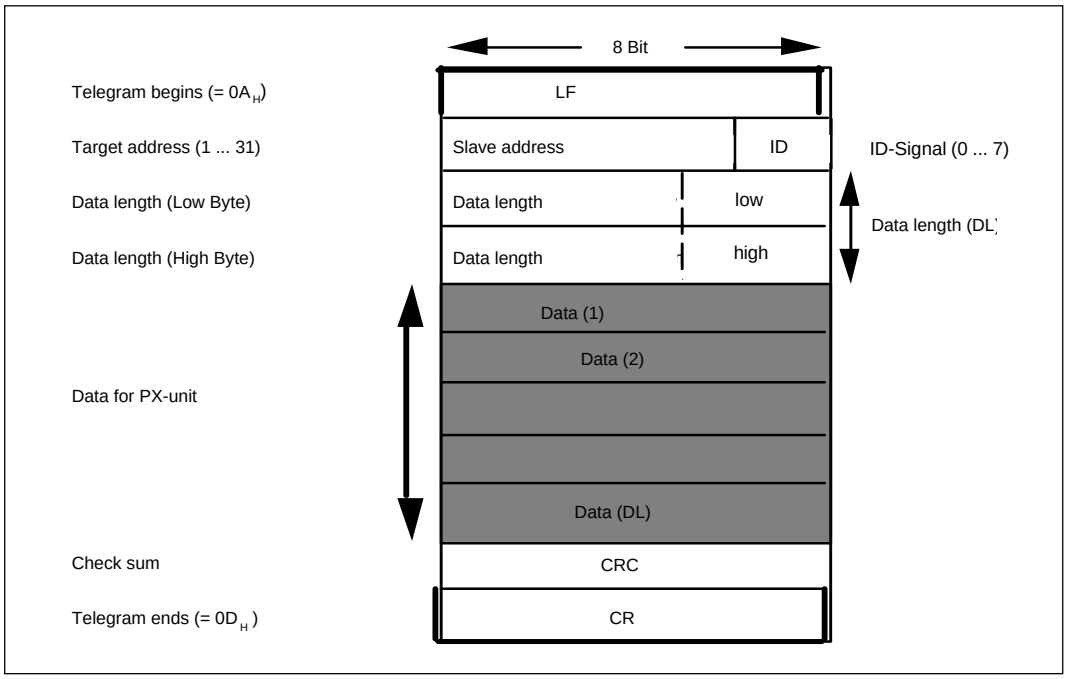
Serial Drive
Display Mode

Display mode is active when the unit is switched on.

Information from the controller is transmitted to the display in send telegrams. Each send telegram is followed by a receive telegram, in which the text display responds with a positive or negative acknowledgement.

Data Communication Protocol

With serial drive, the following data communication protocol must be adhered to:



The text display address (1 ... 31) should be coded in binary form in the 5 bits of the **Slave address**. This address must correspond to the unit-address configured when the text memory was programmed.

If the **ID signal** = 7, all the displays connected to the network will be called up, but only the display with the transmitted slave address will respond. If the ID ≠ 7 (0 ... 6), there will be no effect on the operation of the text display(s). The **slave address/ID-Byte** will be sent back unchanged in the response telegram.

In the Bytes **data length low** and **data length high**, all the data for the PX-unit must be contained in byte form between **Data (1)** and **Data (DL**).

The operating mode should be coded in **Data (1)**.

Depending on which mode is selected, additional information should be transmitted in the following bytes **Data (2)** to **Data (DL)** (see section entitled "Operating Modes", from page 3-23 onwards).

The check sum CRC should be calculated as described in the section entitled "Example for calculating check sum CRC" on page 3-30.

The PX-unit which was addressed will respond with a positive or negative acknowledgement.

PX 30/PX 120 Response

The PX-unit which was addressed will respond to the telegram as follows:

- OAH **ID** 01H 00H 06H **CRC** 0DH — Positive acknowledgement (06H)
- 0AH **ID** 01H 00H 15H **CRC** 0DH — Negative acknowledgement (15H); Transmission error
- OAH **ID** 04H 00H 15H 00H xx xx **CRC** 0DH (xx xx ... 16-bit error number) — Negative acknowledgement (15H); Error which triggers error message "ERROR xxx" in the PX-unit (xxx ... decimal error number; see section entitled "Error Codes" on page 7-4 of the Appendix)

A positive acknowledgement indicates that the text display has received the data without error and has carried out the command. A negative acknowledgement indicates that the data has not been received without error and/or the command has not been carried out.

Operating Modes

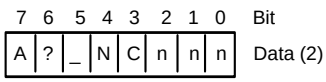
With the exception of **Auto Display of text memory contents on/off and Status display on**), the same operating modes and special functions are available when driven in serial as in parallel. Data communication must be carried out in accordance with the protocol described above.

Important: Binary-coded values are interpreted as signed integers in serial drive as well (see page 3-4)!

The operating modes available are:

Operating Mode	Hex-Code in Data (1)	Binary Code in Data (1)
Text Mode	50	01010000
Variable Mode	60	01100000
Special Mode	70	01110000
Activating Monitor Mode	90	10010000

In "text" and "variable" modes, the ID-Byte in **Data (2)** should be transmitted in the following format:



A = 1 A = 0	Display IBM-ASCII-coded variables Display binary or BCD-coded variables
? = 1 ? = 0	Inserts "?"; flashes alternately with the lowest value digit of the variable value "?" not inserted
_ = 1 _ = 0	Inserts "_"; flashes alternately with the lowest value digit of the variable value "_" not inserted
If ? = 1 **and** _ = 1, "?" will be inserted. "?" or "_" can still be flashed at the point of the lowest value digit even when the variable value is not transmitted (Data (2) as Data (DL)).	
N = 1 N = 0	A negative prefix ("-") is placed before the variable value Display without negative prefix
Important: The "N" Bit is only significant when you are selecting variables in BCD-mode. Binary-mode operates with signed integers.	
C = 1 C = 0	BCD-coded transmission Binary-coded transmission
nnn	In variable mode, variable numbers (0 ... 7) should be transmitted in Bits 2 ... 0.

In special mode, the special function code (see Page 3-11) is selected in Data (2) in bits 5 ... 0.

The following table shows the connection between operating mode, coding and the contents of the data field from **Data (1) to Data (DL)**.

Operating Mode	Data (n)	Contents
Binary-coded text call-up	Data (1)	50_H
	Data (2)	ID-Byte: Binary xxxx**0**xxx (eg. 00_H) (x ... Bit is insignificant)
	Data (3)	low Byte text number
	Data (4)	high Byte text number
BCD-coded text call-up	Data (1)	50_H
	Data (2)	ID-Byte: Binary xxxx**1**xxx (eg. 08_H) (x ... Bit is insignificant)
	Data (3)	BCD-digit 10^1, 10^0 of the text number
	Data (4)	BCD-digit 10^3, 10^2 of the text number (Data (4) is omitted for text numbers 1 ... 99)

Operating Mode	Data (n)	Contents		
Binary-coded variable call-up	Data (1)	60_H		
	Data (2)	ID-Byte: binary **0**(?)(_)x**0**nnn (nnn ... variable number binary coded, x ... Bit is insignificant)		
	Data (3)	8-bit-binary value	16-bit binary value	32-bit binary value
	Data (4)			
	Data (5)			
	Data (6)			
		Data (4) to Data (6) or Data (5) and Data (6) may be omitted		
BCD-coded variable call-up	Data (1)	60_H		
	Data (2)	ID-Byte: binary **0**(?)(_)(N)**1**nnn (nnn ... variable number binary coded)		
	Data (3)	BCD-value 10^1, 10^0		
	Data (4)	BCD-value 10^3, 10^2		
	Data (5)	BCD-value 10^5, 10^4		
	Data (6)	BCD-value 10^7, 10^6		
	Data (7)	BCD-value 10^9, 10^8		
		Data (4) ... Data (7) may be omitted.		
ASCII-coded variable call-up	Data (1)	60_H		
	Data (2)	ID-Byte: binary **1**(?)(_)(N)xnnn (nnn ... variable number binary coded; x ... Bit is insignificant)		
	Data (3)	1st ASCII-character (furthest **right**)		
	•	•		
	•	•		
	•	•		
	Data (12)	10th ASCII-character (furthest **left**)		
		Data (4) ... Data (12) may be omitted.		
Special Mode	Data (1)	70_H		
	Data (2)	binary xx**yyyyyy** yyyyyy Special function code, binary coded 000001 Segment test on 000010 Segment test off 000011 Output version number 000110 Clear display (See Page 3-11), x ... Bit is insignificant.		
Activating Monitor Mode	Data (1)	90_H		

Examples for Serial Drive in Display Mode

Example 1: BCD-coded text call-up
Send Text Number 1

Telegram:

Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0 (Bit 2^7 ... 2^3: Unit address in binary code / Bit 2^2 ... 2^0: ID-Signal)
3	03_H	Amount of data for PX-unit in Bytes (Low Byte)
4	00_H	Amount of data for PX-unit in Bytes (High Byte)
5	50_H	1st data byte: Code for operating mode (50_H = text mode)
6	08_H	2nd data byte: Coding of text number (08_H = BCD-coded)
7	01_H	3rd data byte: Decade 10^1 and 10^0 of text number
8	$9C_H$	Check sum (CRC) of bytes 2 to 8
9	$0D_H$	End of telegram

Example 2: Binary-coded text call-up
Send Text Number 19

Telegram:

Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0
3	03_H	Amount of data for PX-unit in Bytes (Low Byte)
4	00_H	Amount of data for PX-unit in Bytes (High Byte)
5	50_H	1st data byte: Code for operating mode (50_H = text mode)
6	00_H	2nd data byte: Coding of text number (00_H = binary-coded)
7	13_H	3rd data byte: Text number (Low Byte)
8	92_H	Check sum (CRC) of bytes 2 to 8
9	$0D_H$	End of telegram

Example 3: Variable transmission, BCD-coded
Insert the value 1 234 567 890 for variable 1 in the text masks called up using examples 1 or 2.

Telegram:

Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0
3	07_H	Amount of data for PX-unit in Bytes (Low Byte)
4	00_H	Amount of data for PX-unit in Bytes (High Byte)
5	60_H	1st data byte: Code for operating mode (60_H = variable mode)
6	09_H	2nd data byte: Variable 1, numeric value, positive, BCD-coded
7	90_H	3rd data byte: Decade 10^1 = 9 and Decade 10^0 = 0
8	78_H	4th data byte: Decade 10^3 = 7 and Decade 10^2 = 8
9	56_H	5th data byte: Decade 10^5 = 5 and Decade 10^4 = 6
10	34_H	6th data byte: Decade 10^7 = 3 and Decade 10^6 = 4
11	12_H	7th data byte: Decade 10^9 = 1 and Decade 10^8 = 2
12	$E4_H$	Check sum (CRC) of bytes 2 to 11
13	$0D_H$	End of telegram

Example 4: Variable transmission, binary-coded
Insert the value 18 for variable 1 in the text masks called up using examples 1 or 2.

Telegram:

Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0
3	03_H	Amount of data for PX-unit in Bytes (Low Byte)
4	00_H	Amount of data for PX-unit in Bytes (High Byte)
5	60_H	1st data byte: Code for operating mode (60_H = variable mode)
6	01_H	2nd data byte: Variable 1, numeric value, binary-coded
7	12_H	3rd data byte: Priority 2^7 ... 2^0
8	82_H	Check sum (CRC) of bytes 2 to 7
9	$0D_H$	End of telegram

Example 5: Variable transmission, ASCII-coded
Insert the character sequence "ABCDEFGHIJ" for variable 7 in the text masks called up using examples 1 or 2.

Telegram:

Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0
3	$0C_H$	Amount of data for PX-unit in Bytes (Low Byte)
4	00_H	Amount of data for PX-unit in Bytes (High Byte)
5	60_H	1st data byte: Code for operating mode (60_H = variable mode)
6	87_H	2nd data byte: Variable 7, ASCII-coded
7	$4A_H$	3rd data byte: 1st character (starting at the right; "J")
8	49_H	4th data byte: 2nd character (left of 1st character; "I")
9	48_H	5th data byte: 3rd character (left of 2nd character; "H")
10	47_H	6th data byte: 4th character (left of 3rd character; "G")
11	46_H	7th data byte: 5th character (left of 4th character; "F")
12	45_H	8th data byte: 6th character (left of 5th character; "E")
13	44_H	9th data byte: 7th character (left of 6th character; "D")
14	43_H	10th data byte: 8th character (left of 7th character; "C")
15	42_H	11th data byte: 9th character (left of 8th character; "B")
16	41_H	12th data byte: 10th character (left of 9th character; "A")
17	$4E_H$	Check sum (CRC) of bytes 2 to 16
18	$0D_H$	End of telegram

Example 6: Select monitor mode

Telegram:

Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0
3	01_H	Amount of data for PX-unit in Bytes (Low Byte)
4	00_H	Amount of data for PX-unit in Bytes (High Byte)
5	90_H	1st data byte: Code for activating monitor mode
6	67_H	Check sum (CRC) of bytes 2 to 5
7	$0D_H$	End of telegram

Example 7: Positive Acknowledgement
Telegram has been received by the PX-unit and the command carried out.

Telegram:

Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0
3	01_H	Amount of data containing back information in Bytes (Low Byte)
4	00_H	Amount of data containing back information in Bytes (High Byte)
5	06_H	1st data byte: Positive acknowledgement
6	$F1_H$	Check sum (CRC) of bytes 2 to 5
7	$0D_H$	End of telegram

Example 8: Negative acknowledgement
Telegram contained a text number under which no text was edited:
(⟶ ERROR 30: no text available; record unoccupied)

Telegram:

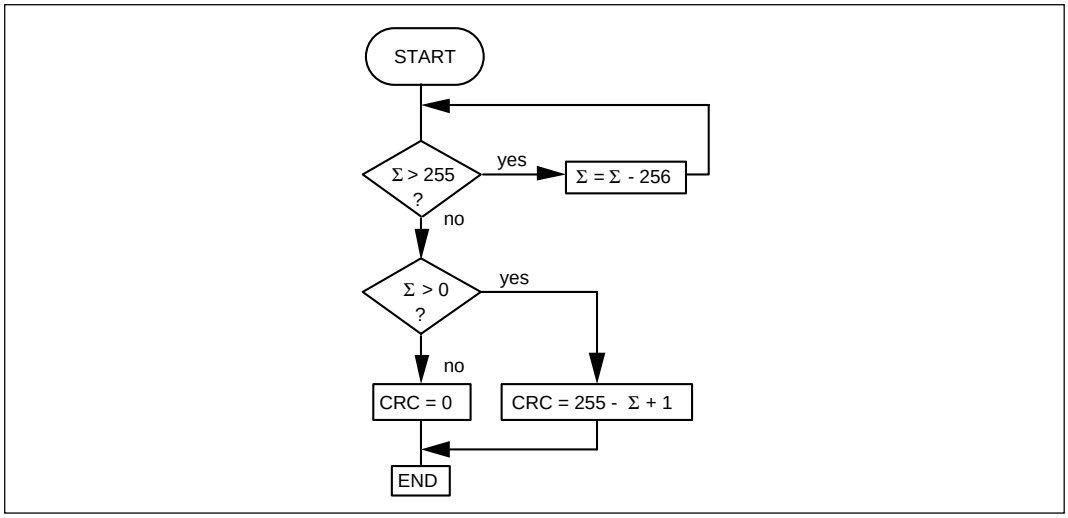
Byte		
1	$0A_H$	Start of telegram
2	08_H	Address 1 / ID-Signal 0
3	04_H	Amount of data containing back information in Bytes (Low Byte)
4	00_H	Amount of data containing back information in Bytes (High Byte)
5	15_H	1st data byte: Negative acknowledgement
6	00_H	2nd data byte
7	$1E_H$	3rd data byte: Error number 30 (Low Byte)
8	00_H	4th data byte: Error number 30 (High Byte)
9	$C1_H$	Check sum (CRC) of bytes 2 to 8
10	$0D_H$	End of telegram

Please note: When data is not received correctly (transmission error), Bytes 6, 7 and 8 will be missing from the PX-unit's response telegram (Byte 3 = 01_H and CRC-Byte 9 = $E2_H$).

Example for calculating the check sum CRC
Example 1 (BCD-coded text call-up) on page 3-26 shows how the check sum CRC is calculated.

First of all, calculate the sum total (Σ) of all bytes from byte 2 (slave address / ID-Byte) up to and including the last data byte (Data (DL); Byte 7 in the example).

To calculate the check sum CRC, follow the procedure depicted below, using the sum total (Σ).



In Example 1:

	Byte 2	08_H
	Byte 3	03_H
	Byte 4	00_H
	Byte 5	50_H
	Byte 6	08_H
	Byte 7	01_H
	Σ =	$64_H = 100_D$

$CRC = (255 - 100 + 1)_D = 156_D$
$CRC = (FF - 64 + 01)_H = 9C_H$

Text Display Address and ID-Signal

The text display address and ID-Signal are transmitted in Byte 2 of the send telegram (slave address/ID-Byte). Coding is as follows:

	Text display address					ID-Signal			
	7	6	5	4	3	2	1	0	Slave address Bit / ID-Byte
Address 1	0	0	0	0	1	0	0	0	ID-Signal 0
Address 2	0	0	0	1	0	0	0	1	ID-Signal 1
.	.	.	.	.	.	.	.	.	.
Address 31	1	1	1	1	1	1	1	1	ID-Signal 7 = Global Call

Serial Drive
Monitor Mode

Display mode is active when the unit is switched on. If several PX-units are networked together, they cannot be operated in monitor mode.

To activate monitor mode you will need to send the following telegram:

Start byte	$0A_H$
Slave Address / ID	
Data length, low byte	01_H
Data length, high byte	00_H
Data (1)	**90_H**
CRC	
End byte	$0D_H$

Communication Protocol

1-byte data pieces are processed in monitor mode. These are either:

- Representable characters from the IBM character set, which will be displayed at the current cursor position OR
- Characters with control functions (control codes).

The control codes are listed on pages 7-3/7-4.

The 1-byte data pieces can be sent from the host to the PX-unit either individually or in a package. The text display will respond to each of these send telegrams with a positive or negative acknowledgement.

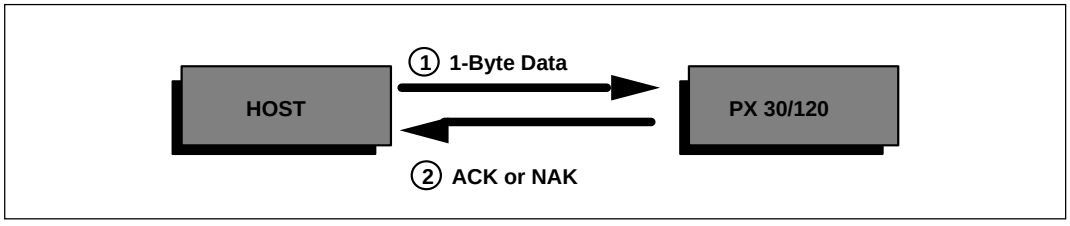
Communication Protocol for individual data

Host:

1-byte data (IBM/ASCII)

PX 30/120:

positive acknowledgement:	06H (ASCII: ACK)
negative acknowledgement:	15H (ASCII: NAK)



Communication Protocol for packaged data

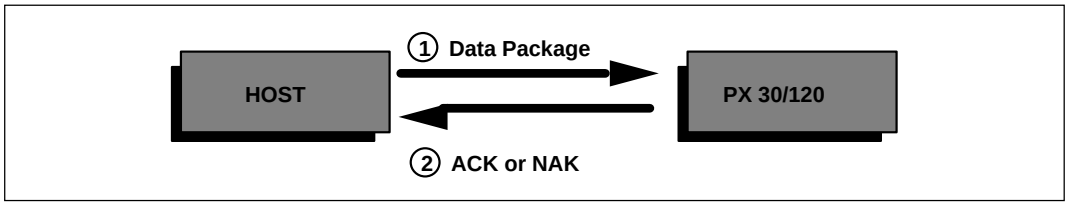
Data can also be transmitted in a package, speeding up data communication between the host and the PX-display. Confirmation will only be given for one data package.

Host:

Start byte:	02H
Data (1):	IBM/ASCII-characters: representable characters and control codes
.	
.	
.	
Data (n):	
End byte:	03H

PX 30/120:

positive acknowledgement:	06H (ASCII: ACK)
negative acknowledgement:	15H (ASCII: NAK)



When the display is in monitor mode, control codes can be used to switch between the IBM and Cyrillic character set (see section entitled "Control Codes" in the Appendix).

Text in Display Mode

Text Memory

The text memory is fitted with a Flash-EPROM with a 32 KByte capacity. Its address range consists of record numbers 1 ... 9999. The address range has been made this large so that records can be numbered both sensibly and systematically.
EPROMs with a 32KByte memory capacity and an access time of $\leq$ 250 ns may also be used, however.

Creating Text

The software required for creating text and programming the text memory is described in detail in the "ADIT DOS" Manual.

Character Set

Both the IBM and Cyrillic character sets can be used on the PX 30 and PX 120.
Please note that characters $1B_H$ and 23_H (#) are used for special functions.

Character (hex)	Function
1B	Used as a start character for an ESC-Sequence; ESC-Sequences determine the attributes of text and variables (eg. flashing characters) and also clear the display before showing new text (see section below entitled "Control Characters")
23	Text mask for variables (#)

The range of characters that can be represented is from 20_H to FE_H.

A configuration instruction can be used to switch to the Cyrillic character set (see the section entitled "Configuration").

Control Characters (ESCAPE Sequences)

The following control characters are supported by the PX 30 and the PX 120:

Control Command	ESC-Sequence	Explanation
Clear display	ESC 40	Displayed text is cleared. Text attributes (variables etc.) are also lost.
Flasher on	ESC 06	From this point on, text will flash. This does not affect the variables.
Flasher off	ESC 07	From this point on, text will no longer flash.
Line Break	ESC 45	Corresponds to the ASCII: CR-LF
Right Justification *	ESC 56	Refer to page 5-2, "Variable Layout"

Text in Display Mode

Control Command	ESC-Sequence	Explanation
Left Justification	ESC 57	Refer to page 5-2, "Variable Layout"
Show leading zeros	ESC 58	Refer to page 5-2, "Variable Layout"
Suppress leading zeros	ESC 59	Refer to page 5-2, "Variable Layout"
Variable No. 0	ESC 60	
Variable No. 1	ESC 61	
•	•	
•	•	
Variable No. 7	ESC 67	

***** Default Setting: This function has priority. Even if the relevant control character is not present, this function will remain in effect until the opposite function is set.

Layout

When a text is called up, control characters contained in the text are evaluated and then converted into the corresponding text and variable layout (see "Control Characters").

Please note: If you are using ADIT DOS Software, these control characters are selected as a "Language Command" in the dialogue; the relevant ESCAPE Sequence is then generated automatically and is inserted in the text at the cursor position.

The layout of text is line-oriented, ie. if a text is to carry over on to the next line, the corresponding control character must be edited as such.

In the editor section of "ADIT DOS", line breaks (carriage return and line feed) occur automatically.

Clear Display

If a text contains the control character "clear display", the previous text and all the corresponding variable positions will be cleared when a new text is displayed.

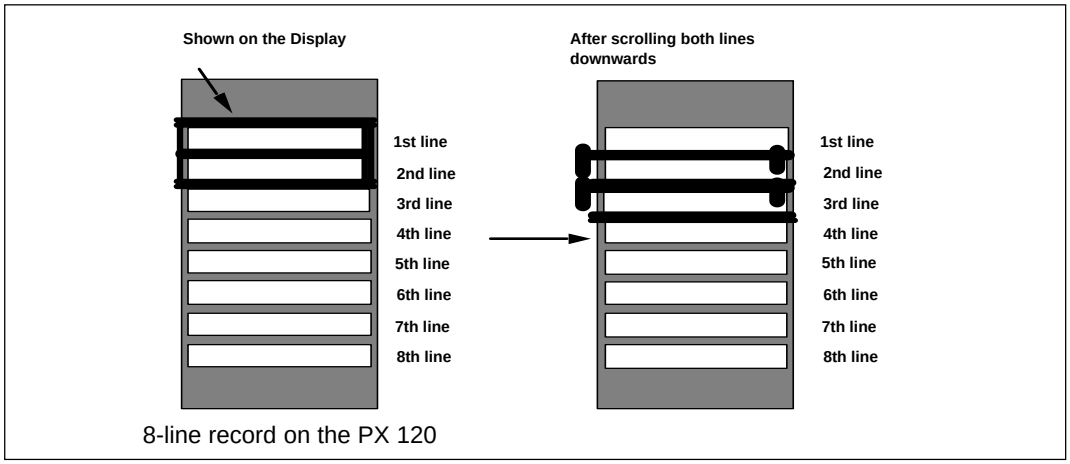
If the "clear display" command has been deleted in a record in the ADIT DOS Editor, this record can be "stored" in a type of "background file" on top of the previously displayed text. All the edited characters in the "new" text (incl. spaces) overwrite the characters they replace in the "old" text. Any characters in the "old" text that are passed over using the arrow keys (in ADIT DOS: "." character), will be retained. Variable positions are retained and may also be addressed. If the "old" text is scrolled, the "new" text (also scrolled" will be placed on top.

Scrolling Background Text

It is possible to edit records with more than two lines of text. A record may contain a maximum of 320 characters (without control characters). On the PX 30 (20 characters per line), 16 lines of text are possible per record and on the PX 120 (40 characters per line), 8 lines. When a record is displayed, the first two lines of the text are shown. In both parallel and serial drive, the scrolling of texts is only possible via the special scroll inputs:

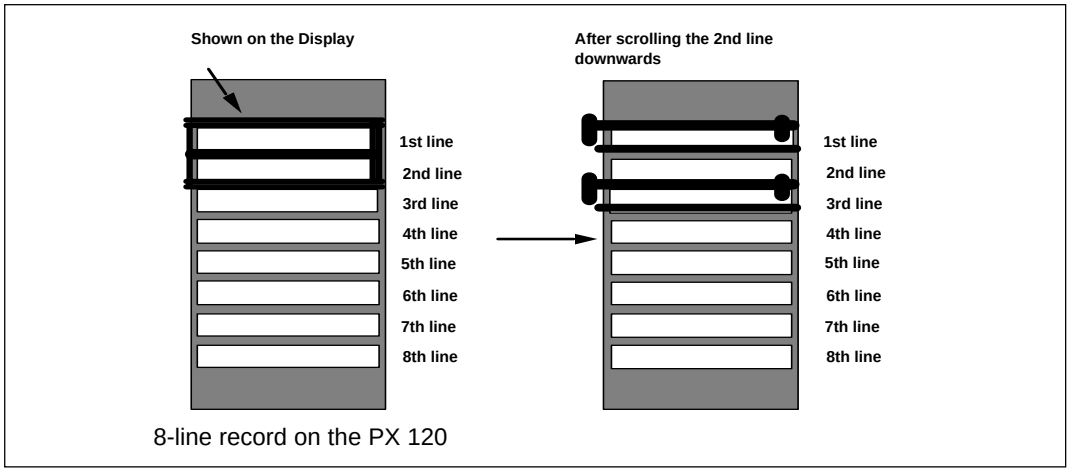
- PX 30: E9 (up) / E8 (down)
- PX 120: E17 (up) / E16 (down)

Additional lines (background text) can be displayed by pressing the keys connected to these scroll inputs. Each time a key is pressed (positive-going pulse edge at the scroll input), the text is scrolled on one line (ie. when scrolling downwards, for example, the 2nd line will become the 1st line on the display and the 3rd line will become the 2nd).



8-line record on the PX 120

A configuration instruction can be used to change the scroll function so that only the 2nd line of the display is scrolled with background text and the 1st line remains on the display (see the chapter entitled "Configuration" on page 6-1).



8-line record on the PX 120

The display will return to the start of the text when both scroll inputs are set simultaneously ie, the first two lines will be displayed.

Notes

Variables in Display Mode

Basics

- 8 variables are available per text (record), with the numbers 0 ... 7.
- A variable can be a maximum of 10 characters in length.
- If the variable consists of figures, the negative prefix "-" may be placed in front of the figure as an 11th character.
  Please note: If a variable is composed of both positive **AND** negative numeric values, with figures after the decimal point, right justification should be selected, otherwise the variable value could be falsified.
- Text masks (#) must be created in the text for variables. These format the variable output.
- If the transmitted variable is larger than the variable length edited, (number of text masks), only the higher value or lower value part will be displayed, depending on the justification (left/right).
- The text display is line-oriented! This means, for example, that where the PX 30 has 2 lines of 20 characters and the ADIT DOS Editor has a default setting of 40 characters per line, all characters edited after the 20th column will not be displayed. It also means that a variable cannot extend over the start or end of a line. For this reason it is not possible to insert one and the same variable over several lines.
- In parallel drive, if a variable value is transmitted in more than one cycle, the first value transmitted is taken as the higher value part.
- In serial drive, if a variable value is transmitted in more than one piece of data, the first value transmitted is taken as the lower value part.

Variable Layout

The following formatting options are available:

- Formatting with the aid of variable masks (ASCII: 23_H, #)
- Left or right justification
- Display or suppress leading zeros

Each digit in a variable must have a "#"-character as a text mask so that it can be fully displayed. The control character for the variable number must be placed at the last "#"-position (see ADIT DOS Manual).

Please note: The control character 23_H (#) is only visible in the ADIT DOS Editor.

Variables in Display Mode

Text Characters within Variables

The variable only appears in the text at the places marked "#". All other text characters are placed within the variable number.

Examples:

Edited Text	Variable Value	Display
#-#-#-#	1234	1-2-3-4
#,##	156	1,56
####/# (show leading zeros)	00105	0010/5

Overlapping Variables

Variables may not overlap. For reasons of operational security, it is not possible to edit two or more variables at the same position.

In the following example, variable b is restricted by variable a:

Edited Text	Variable Value	Display
####,##### a b	a 987654 b 111111	9876,54111

Left / Right Justification

If no control character is present the variable will be displayed with right justification.

Example:

Edited Text	Variable Value	Display
####,##	12345678	Right justified: 3456,78 Left justified: 1234,56

Leading Zeros

If no control character is present the variable will be displayed with leading zeros suppressed.

Examples:

Edited Text	Variable Value	Control Character	Display
#####,## (right justified)	-34	Show leading zeros	-0000,34
		Suppress leading zeros	-0,34

Edited Text	Variable Value	Control Character	Display
#####,## (left justified)	-34	Show leading zeros Suppress leading zeros	-34 , -34 ,
#### (left justified)	0	Suppress leading zeros	0
##,## (right justified)	1	Suppress leading zeros	0,01
#### (left justified)	34	Suppress leading zeros	34
###,## (right justified)	-1	Suppress leading zeros	-0,01
###.## (right justified)	-1	Suppress leading zeros	-0.01
###-## (right justified)	-1	Suppress leading zeros	--1

When leading zeros are suppressed, "superfluous" zeros are replaced by spaces and any negative prefix is moved along accordingly.

Inserting a Cursor or Question Mark

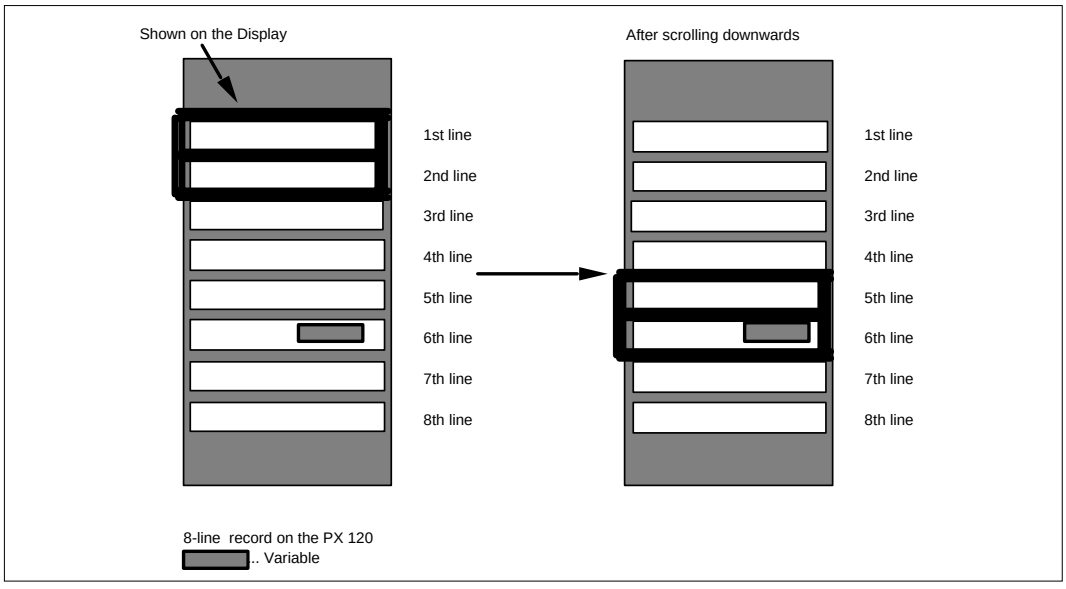
In parallel drive, if inputs "?" (E6) or "_" (E5) are set in the first cycle of the variable selection, the last character (lowest value part) and the question mark ("?") or underline ("_") will flash alternately when the variable value is displayed. This type of flashing display may be used as an operator prompt, if required.

Example: 12345 alternating with: 1234?

If E6 and E5 are set simultaneously, the question mark has priority.
"?" or "_" may be flashed even when there is no variable transmission.

Variables in Background Text

Variables can also be inserted into background text, becoming visible when you scroll to the corresponding line.



Notes

Configuration

On the PX 30 and PX 120, a variety of settings can be made in respect of:

- Coding
- The number of digits transmitted in parallel in BCD-Mode
- The number of inputs used **(on PX 120 only)**
- Scroll function
- Text display address
- Character set
- Default text on power-up.

Setting	Explanation	Default Setting
Coding	Data transmission is either binary or BCD-coded	Binary-coded
Number of digits transmitted in parallel in BCD-Mode	The number of BCD-digits per cycle can be established by means of a configuration instruction. Advantage: You can reduce the number of inputs to the number required for the application, thereby saving PLC outputs. The number of BCD-digits transmitted in parallel is geared to the number of data inputs connected (4/8/12/16). If only 4 data inputs are connected, it will not be possible to transmit ASCII-coded variables!	PX 30: 2 digits PX 120: 4 digits
Number of inputs used **(on PX 120 only)**	You can select between 24 and 16-input mode. If 16-input mode is selected, inputs E8 ... E15 will have no function. Advantages: One PLC-program to drive 16 inputs is sufficient for applications in which either a PX 120 or PX 30 can be used. You will also save 8 PLC outputs.	24-input mode
Scroll function	You can select between scrolling both lines of the display or just the second line. You can scroll upwards as far as the first line and downwards as far as the last line. If both scroll inputs are driven simultaneously the display will return to the start of the text.	Both lines are scrolled
Text display address	Addresses 1 to 31 can be set. Address 0 will automatically be changed to Address 1.	Address 1
Character Set	You can select either the IBM or the Cyrillic character set.	IBM character set
Default text on power-up	The default message which appears on the display after power-up can be replaced by another message created by the user.	"PX 30, Version x.xx" or "PX120, Version x.xx"

Configuration

The settings for coding, number of BCD-digits per transmission, number of inputs used on the PX 120, scroll function, display address and character set are carried out in a single command:

$M:x/y/z/aa/b

x	Coding and number of digits transmitted in parallel in BCD-mode
y	Number of inputs used on the PX 120
z	Scroll function
aa	Text display address
b	Character set

All the characters in this command must be represented.

The $M-configuration instruction (command for changing settings) should be entered in ADIT DOS under the menu heading "Default Settings", in the line "Max. 40 additional characters:"

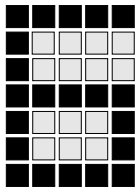
You can also change the default text after power-up in the ADIT DOS default settings, simply be entering the text number of the replacement text in the line "Replacement text number for default setting (0 ... 9999):".

The following table illustrates which options are available for changing the settings and the corresponding command:

Setting	Options	Command
Coding and number of BCD-digits transmitted in parallel	Binary: x = 0 BCD 1 digit (10^9 ... 10^0): x = 1 BCD 2 digits (10^9 ... 10^0): x = 2 BCD 3 digits (10^9 ... 10^0): x = 3 * BCD 4 digits (10^9 ... 10^0): x = 4 * BCD 1 digit (10^0 ... 10^3): x = 5 ** * only on PX 120 ** only in text mode	$M:**0**/y/z/aa/b $M:**1**/y/z/aa/b $M:**2**/y/z/aa/b $M:**3**/y/z/aa/b $M:**4**/y/z/aa/b $M:**5**/y/z/aa/b
Number of inputs used (**PX 120**)	24: y = 0 16: y = 1 y is ignored by the PX 30	$M:x/**0**/z/aa/b $M:x/**1**/z/aa/b
Scroll function	Both lines scrolled: z = 0 Just 2nd line scrolled: z = 1	$M:x/y/**0**/aa/b $M:x/y/**1**/aa/b

Setting	Options	Command
Text display address	1 : aa = 01 . . 31 : aa = 31	$M:x/y/z/**01**/b . . $M:x/y/z/**31**/b
Character set	IBM character set: b = 0 Cyrillic character set: b = 1	$M:x/y/z/aa/**0** $M:x/y/z/aa/**1**

With the specially developed "ADIT DOS" software it is possible to undertake these configurations with ease.



Notes

Appendix

Technical Details

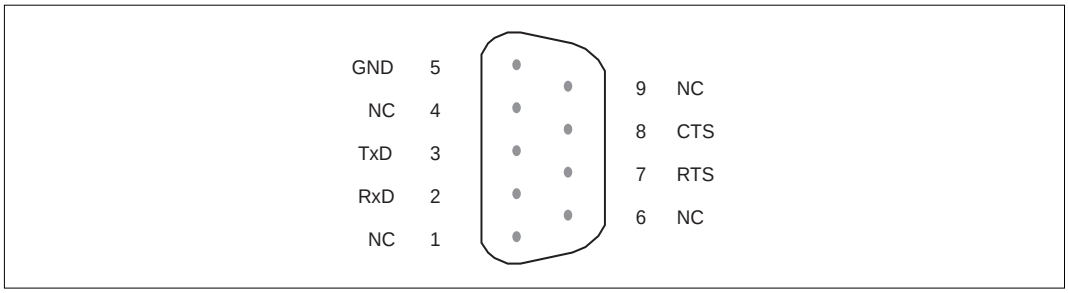
Operating Voltage	18 ...32 VDC
Rated Operating Voltage	24 VDC
Power Consumption	PX 30: average 3.5 W; PX 120: average 5.7 W
Inputs	PX 30: 16 inputs; PX 120: 24 inputs In accordance with VDI 2880 page 2, 24 V=, no galvanic isolation
Outputs	1 output to VDI 2880 page 2, protected against short circuit and overload
Serial Interface	RS 232 (V24), 9-pin D-Sub-connector
Text Memory	32 KByte Flash-EPROM (standard); 64 KByte optional
Interference Resistance	Resistance to breaks in supply voltage: up to 20 ms; Protection against overload: asymmetric up to 2 kV on 24 V supply, data, I/O and signal lines (RS 232 standard) Electrostatic discharge: 8 kV
Operating Temperature	0 ... +55^{0}C
Storage Temperature	-20 ... +70^{0}C
Climatic Conditions	DIN 40040 Class F
Housing and Terminal Protection	Front to IP 65; Text display mounted without seal: IP 50; Text display mounted with seal: IP 64; Rear: IP 20
Display	Fluorescent display, alphanumeric, IBM character set; PX 30: 5 x 7 dot matrix; PX 120: 5 x 12 dot matrix;
Character Height	PX 30: (2 lines of 20 characters): 5 mm PX 120 (2 lines of 40 characters): 9 mm
Weight	PX 30: approx. 0.7 kg; PX 120: approx. 1.4 kg;

Connector Pin Assignment (V24)

PX 30 and PX 120 Text Displays can be driven serially through the V24 interface. The interface parameters are set at:

- **9600 baud**
- **8 data bits**
- **1 stop bit on receive / 2 stop bits on reply**
- **no parity**

Connection to the serial interface (V24-standard) is via a 9-pin SUB-D connector on the rear of the unit. Pin layout is shown in the diagram below:

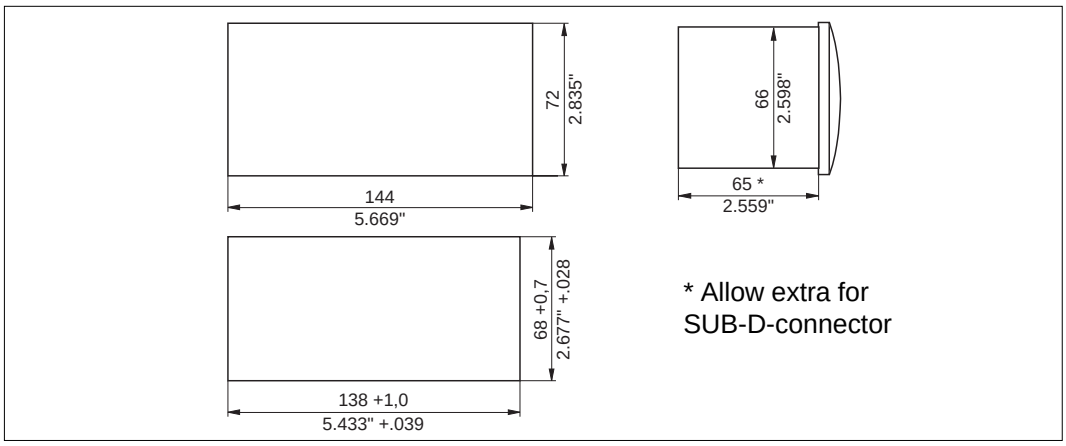


Appendix

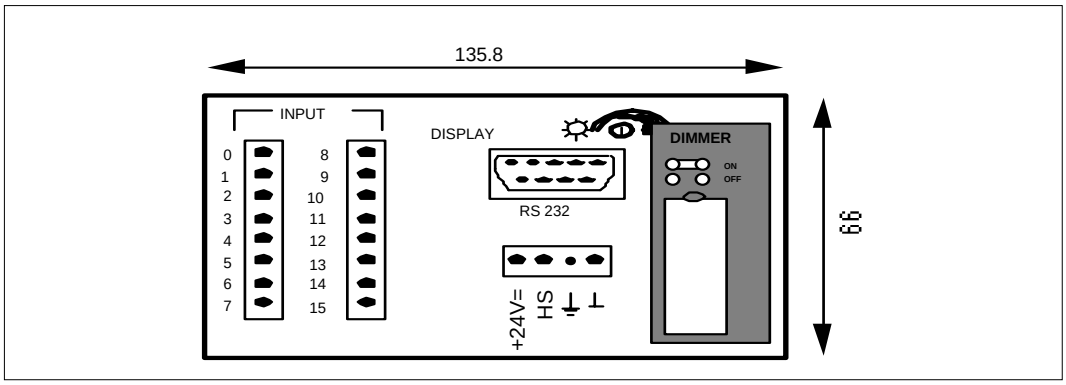
Dimensions

PX 30 Text Display

Dimensions for the front plate and the modular opening are in accordance with DIN 43 700 (dimensions in mm).

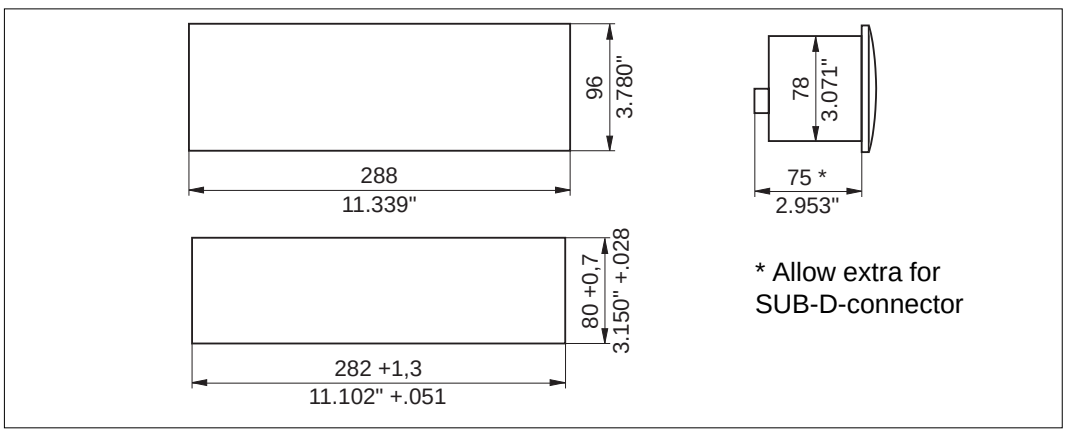


Dimensions for rear of housing (in mm).

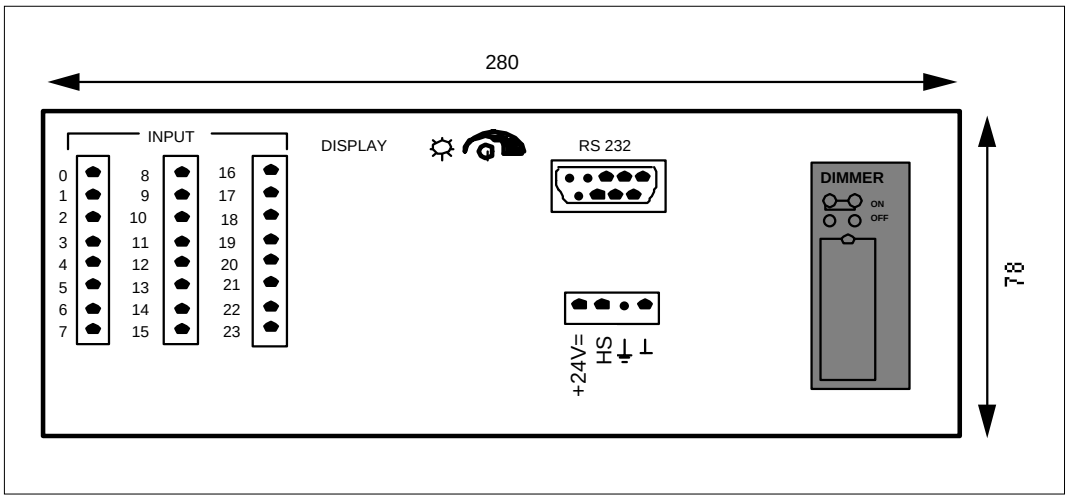


PX 120 Text Display

Dimensions for the front plate and the modular opening (dimensions in mm).



Dimensions for rear of housing (in mm)



Control Codes

Some of the characters from the IBM character set that cannot be represented (range 00_H to $1F_H$) are used as control codes for control commands in monitor mode. The table below shows which characters have which function.

Hexadecimal	Decimal	CTRL-Sequence	Function in Monitor Mode
00	0	^ @	Reserved
01	1	^ A	Mask
02	2	^ B	Start byte for data package
03	3	^ C	End byte for data package
04	4	^ D	Switch on display mode (with serial operation only)
05	5	^ E	Reserved
06	6	^ F	Positive Acknowledgement
07	7	^ G	Move cursor one place to the left
08	8	^ H	Reserved
09	9	^ I	Move cursor one place to the right
0A	10	^ J	Move cursor down one line
0B	11	^ K	Move cursor up one line
0C	12	^ L	Move cursor to start of first line
0D	13	^ M	Move cursor to start of line
0E	14	^ N	Cursor invisible (default)
0F	15	^ O	Cursor visible
10	16	^ P	Reserved
11	17	^ Q	Clear display, cursor invisible
12	18	^ R	Scroll to line n + 1
13	19	^ S	Scroll to start of text
14	20	^ T	Scroll to line n - 1

Appendix

Hexadecimal	Decimal	CTRL-Sequence	Function in Monitor Mode
15	21	^ U	Negative acknowledgement
16	22	^ V	Reserved
17	23	^ W	Reserved
18	24	^ X	Reserved
19	25	^ Y	Reserved
1A	26	^ Z	Set IBM character set
1B	27	^ **[**	Set Cyrillic character set
1C	28	^ ****	Flash "on"
1D	29	^ **]**	Flash "off"
1E	30	^ **^**	Reserved
1F	31	^ _	Reserved

System Messages

- After power-up: "PX 30, Version x.xx" / "PX 120, Version x.xx" (x.xx ... version no.), or any text configured by the user.
- When programming your Flash EPROM: "F-EPROM" flashes. This message disappears when the next action is taken (eg. text call-up, error message)
- When errors occur: "ERROR xxx" flashes (xxx ... error code) or the configured error message.

Error Codes

Error Code (decimal)	**Meaning**
010	Syntax error in configuration instruction
030	No text available; record is not occupied
031	No text available; record is outside the edited text number range
035	Text number is outside the text memory address range (>9999)
040	Too many BCD-digits; max. permitted number of transmission cycles has been exceeded
041	BCD-digit(s) > 9 were found in the text selection
042	Binary value too large
044	Negative text number selected
050	Variable marker has no variable (control character) edited
051	Variable (control character) edited without a marker
052	Too many variable markers # (>11)
080	Special function code invalid (with V24 drive)
081	No variable value in telegram (with V24 drive)
082	No text number in telegram (with V24 drive)

Error Code (decimal)	Meaning
100	Invalid monitor command
200	Clearing error (when programming Flash-EPROMs)
201	ID-code error (when programming Flash-EPROMs)
202	Write error (when programming Flash-EPROMs)
203	Text memory too small (when programming Flash-EPROMs)

Error Messages (from version no. 1.20)

The configuration instruction used to edit the default message after power-up ($B-command) can also be used to input a text number to indicate the start of the text memory range for error messages. Any texts (records) stored from this text number onwards will be assigned error codes (error numbers) directly, ie. the first text in the range will be assigned the error code 000 (ERROR 000), the next will be assigned error code 002 (ERROR 002) etc.

The error message memory range may contain 256 records (including empty records for error codes that are not used). If the value of the first text number for error messages lies between 0001 and 9744, the allocated text will be displayed instead of the ERROR-Code. If no error message has been edited, the message "ERROR xxx" (xxx ... error code) will be displayed.

The $B-command should be added in ADIT DOS under the menu heading "Default Settings" in the line "Max. 40 additional characters:".

Configuration command:

$B:xxxx/**yyyy**

xxxx Number of the replacement text for the on-line base message (default: xxxx = 0000). You should ensure that this number corresponds to the text number inputted as the default setting.

yyyy Text number which starts the text memory range in which error messages are to be edited.

Procedure after Power-up

When the PX 30 / PX 120 is switched on, an address table for the text memory appears, among other things. The greater the number of texts, the longer this process will take. The time delay between power-up and the display being ready is between 0.3 and 3 seconds, during which time the message: "Self-testing" is displayed. You should always take this time delay into account when switching on your machine. The time loss is justified when you consider how much more quickly text can be selected during operation, and how much less memory is taken up in comparison to texts with a fixed start address.

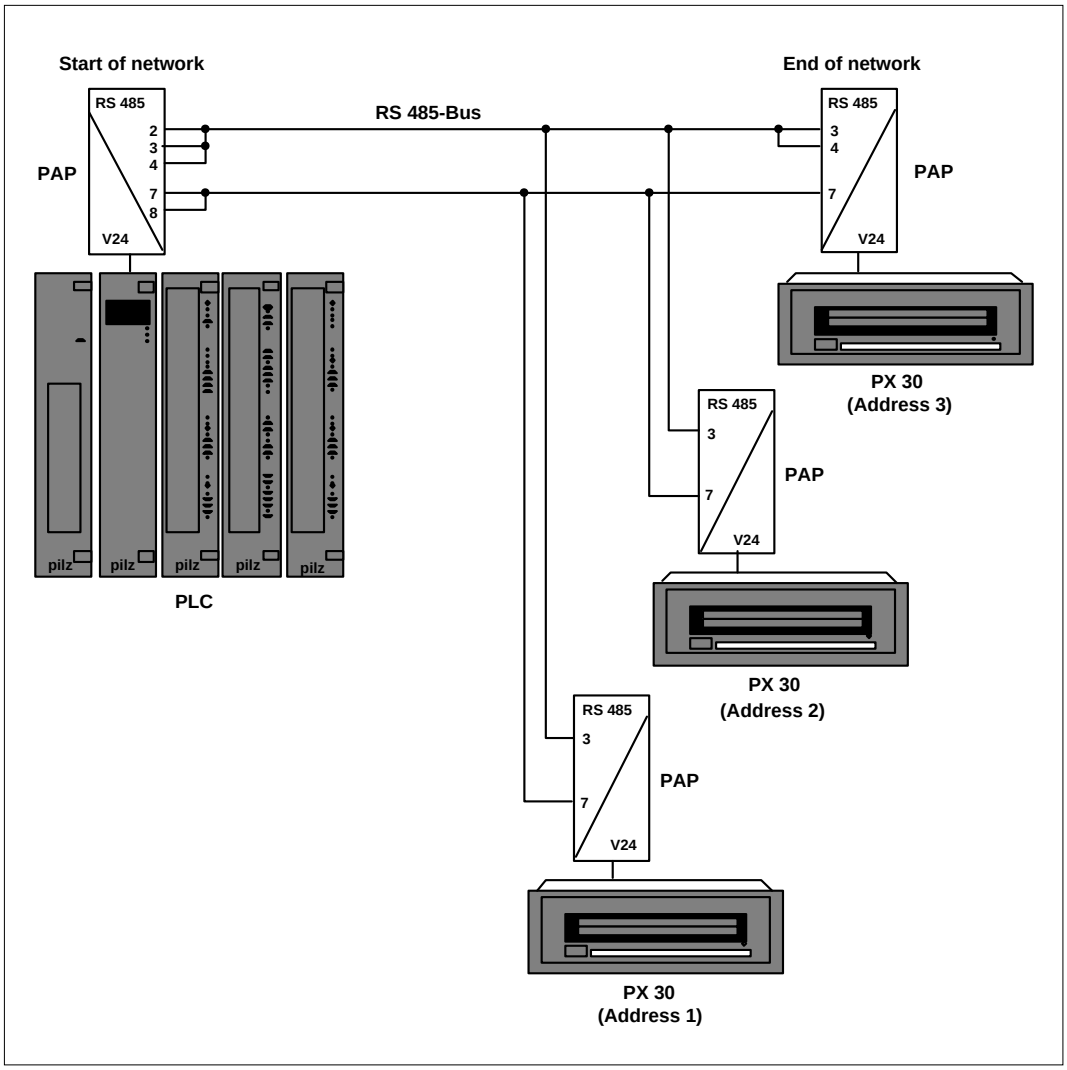
Appendix

Networking Capabilities

Up to 31 text displays (both PX types) can be networked and driven through a PC or PLC. You will need:

- Transmitter with RS 485 interface **OR** a V24 with V24 - RS 485 interface adapter
- One interface adapter (PAP) per PX-unit, RS 485 to V24 standard.

Each text display in the PX-network can then be driven via a configurable display address stored in the text memory.



Example for networking 3 text displays to an RS 485-Bus via a PAP interface adapter.

It is not possible to operate networked displays in monitor mode.

Important! Please ensure that the controller is programmed to switch the interface from send to receive as quickly as possible (within 1 ms) after a telegram is sent.

Character Sets

IBM Character Set

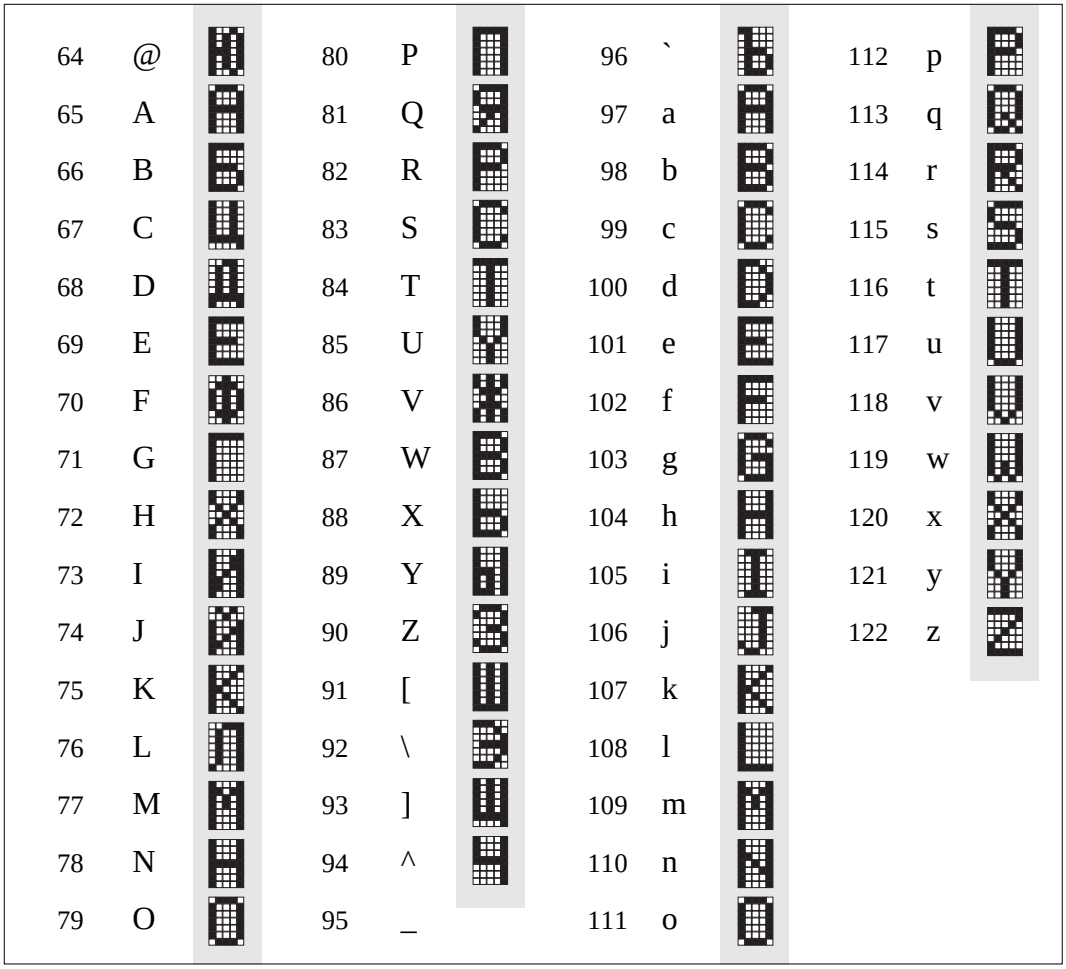
The character set on the PX 30 and PX 120 corresponds to the extended ASCII character set from IBM. Characters 00_H to $1F_H$ (0-31 decimal) are interpreted partly as special control codes in the display's monitor mode (see section entitled "Control Codes" on Pages 52 and 53). The character FF_H (255 decimal) is used to identify an empty text memory.
Characters from the IBM character set that can be represented are 20_H to FE_H (32 to 254 decimal); one exception is the character 35_D (#) which is reserved as a variable marker.

Dec.	Hex.	Char.	Dec.	Hex.	Char.	Dec.	Hex.	Char.	Dec.	Hex.	Char.	Dec.	Hex.	Char.	Dec.	Hex.	Char.	Dec.	Hex.	Char.
32	20		64	40	@	96	60	`	128	80	Ç	160	A0	á	192	C0	└	224	E0	α
33	21	!	65	41	A	97	61	a	129	81	ü	161	A1	í	193	C1	┴	225	E1	ß
34	22	"	66	42	B	98	62	b	130	82	é	162	A2	ó	194	C2	┬	226	E2	Γ
35	23	#	67	43	C	99	63	c	131	83	â	163	A3	ú	195	C3	├	227	E3	π
36	24	$	68	44	D	100	64	d	132	84	ä	164	A4	ñ	196	C4	─	228	E4	Σ
37	25	%	69	45	E	101	65	e	133	85	à	165	A5	Ñ	197	C5	┼	229	E5	σ
38	26	&	70	46	F	102	66	f	134	86	å	166	A6	ª	198	C6	╞	230	E6	µ
39	27	'	71	47	G	103	67	g	135	87	ç	167	A7	º	199	C7	╟	231	E7	τ
40	28	(	72	48	H	104	68	h	136	88	ê	168	A8	¿	200	C8	╚	232	E8	Φ
41	29	)	73	49	I	105	69	i	137	89	ë	169	A9	⌐	201	C9	╔	233	E9	Θ
42	2A	*	74	4A	J	106	6A	j	138	8A	è	170	AA	¬	202	CA	╩	234	EA	Ω
43	2B	+	75	4B	K	107	6B	k	139	8B	ï	171	AB	½	203	CB	╦	235	EB	δ
44	2C	,	76	4C	L	108	6C	l	140	8C	î	172	AC	¼	204	CC	╠	236	EC	∞
45	2D	-	77	4D	M	109	6D	m	141	8D	ì	173	AD	¡	205	CD	═	237	ED	φ
46	2E	.	78	4E	N	110	6E	n	142	8E	Ä	174	AE	«	206	CE	╬	238	EE	ε
47	2F	/	79	4F	O	111	6F	o	143	8F	Å	175	AF	»	207	CF	╧	239	EF	∩
48	30	0	80	50	P	112	70	p	144	90	É	176	B0	▒	208	D0	╨	240	F0	≡
49	31	1	81	51	Q	113	71	q	145	91	æ	177	B1	▒	209	D1	╤	241	F1	±
50	32	2	82	52	R	114	72	r	146	92	Æ	178	B2	▓	210	D2	╥	242	F2	≥
51	33	3	83	53	S	115	73	s	147	93	ô	179	B3	│	211	D3	╙	243	F3	≤
52	34	4	84	54	T	116	74	t	148	94	ö	180	B4	┤	212	D4	╘	244	F4	⌠
53	35	5	85	55	U	117	75	u	149	95	ò	181	B5	╡	213	D5	╒	245	F5	⌡
54	36	6	86	56	V	118	76	v	150	96	û	182	B6	╢	214	D6	╓	246	F6	÷
55	37	7	87	57	W	119	77	w	151	97	ù	183	B7	╖	215	D7	╫	247	F7	≈
56	38	8	88	58	X	120	78	x	152	98	ÿ	184	B8	╕	216	D8	╪	248	F8	°
57	39	9	89	59	Y	121	79	y	153	99	Ö	185	B9	╣	217	D9	┘	249	F9	∙
58	3A	:	90	5A	Z	122	7A	z	154	9A	Ü	186	BA	║	218	DA	┌	250	FA	·
59	3B	;	91	5B	[	123	7B	{	155	9B	¢	187	BB	╗	219	DB	▮	251	FB	√
60	3C	<	92	5C	\	124	7C	¦	156	9C	£	188	BC	╝	220	DC	▄	252	FC	η
61	3D	=	93	5D	]	125	7D	}	157	9D	¥	189	BD	╜	221	DD	▌	253	FD	²
62	3E	>	94	5E	^	126	7E	~	158	9E	₧	190	BE	╛	222	DE	▐	254	FE	■
63	3F	?	95	5F	_	127	7F	▮	159	9F	ƒ	191	BF	┐	223	DF	▀	255	FF	

Appendix

Cyrillic Character Set

When the Cyrillic character set is used, the IBM-characters in the range 64 to 122 decimal (40_H to $7A_H$) are replaced by the characters contained within the square highlighted in the diagram below.

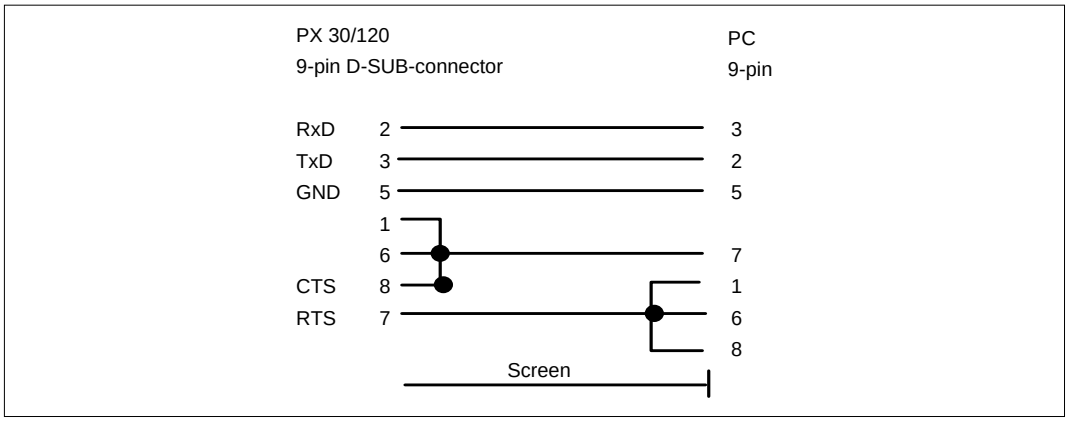


V24 Connection Cable - Pin Layout

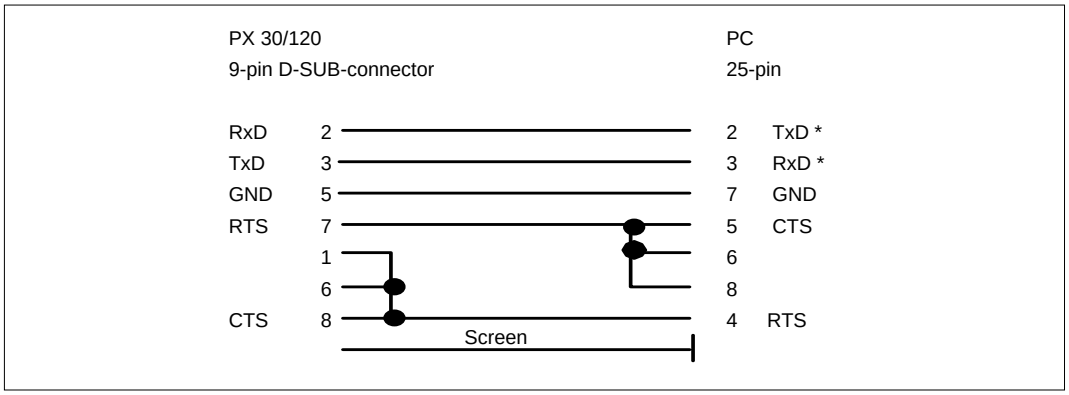
The V24 (RS 232) interfaces must be connected in order to drive the PX 30 and PX 120 in serial and to program the Flash-EPROM directly in the text display. This connection may be made by:

- Ordering the AKSET as an accessory (Order reference 307 426).
  This consists of a zero modem cable (9 to 9-pin) and a mouse adapter (9 to 25-pin) **OR**
- Making up your own cable as shown in either of the two diagrams below.

V24 Connection Cable, 9 to 9-pin



V24 Connection Cable, 9 to 25-pin



* Depending on the type of computer you use, these wires may need to be transposed.

Please note that the V24 interface on the text display uses a 9-pin D-Sub-connector!

Appendix

Notes