topline

Manual PCS 091

for the operating consoles
PCS 009, PCS 090, PCS 095, PCS 095.1

The operating consoles PCS*topline* offer the highest degree on perfection, unparalled in design and function. PCS*topline* keeps everything under control - from the PCS*mini* to the PCS*maxi*, with a superior operating culture and an unlimited setup freedom.

PCS, the first programmable operating console with a large selection of "ready-to-use" operating functions or operating tools which are simply selected via instructions. You can realize even the most unuasal operating requests at ease and in a minimum of time.

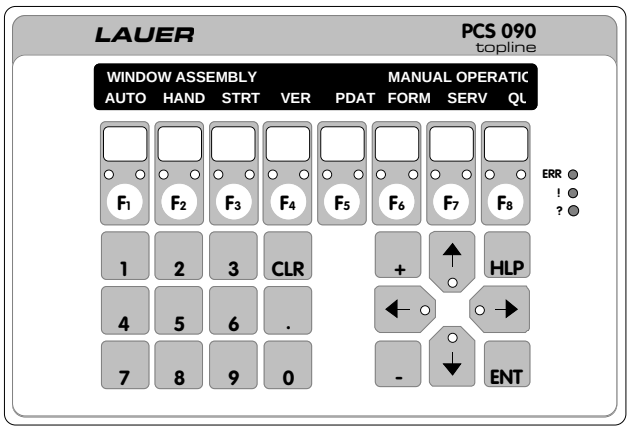
Today this way and tomorrow that way

One standard hardware for virtually thousands of different operating situations. Without extensive wiring and dozens of I/O points.

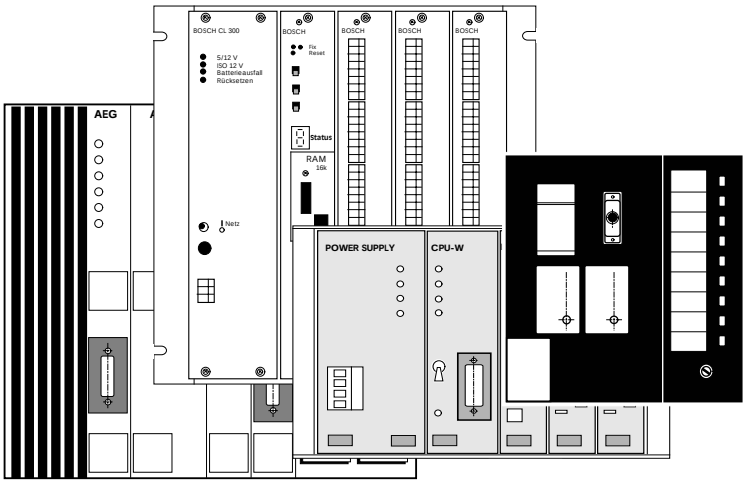
PCS for operating. What else?

- **Machine operation using 8 (PCS 009, PCS 090) or 16 (PCS 095, PCS 095.1) freely assignable keys.** These F01 to F08/F16 labeled keys can be application specifically inscribed and are provided to the controller as status bits.
- **Machine operation using 4 (PCS 009), 16 (PCS 090) or 32 (PCS 095, PCS 095.1) freely usable LEDs.** These can be assigned the indicating states ON, DARK, FLASHING, and INVERSE FLASHING. A green and a yellow LED is allocated to each function key.
- **Display of fixed texts with integrated variable values.** The values can be represented selectably as numerical values or in text format.
- **Representation of the contents of 233 words as variables.** In addition, 650 external variables can be defined. **9 variable formats** (from bit to timer) are available.
- **3 text groups,** 128 operating texts as menu and idle texts, 128 message texts with up to 332 lines, 5 help texts with up to 32 lines.
- **127 menus with 255 menu nodes** each for any menu configurations.
- **4 different deletion modes.** For every message, 1 of 4 possible deletion modes can be selected.
- **Modification of the content of any word within the transfer area.** Using the integrated editor all possible representation formats can be setup.
- **7 priority levels** for idle text up to help text, 3 message priorities Information, Warning, Fault. This working-condition related management significantly off-loads the programmable controller program.
- **Monitoring of rising or falling edges of 128 consecutive bits.** The assignment of texts, the manage-ment of 3 priority levels (Information, Warnings, and Faults), keeping the timely sequence as much as possible, organization of the FIRST MESSAGE, LAST MESSAGE, and CYCLIC DISPLAY, the individually settable deletion behaviour, and the representation formats NORMAL and FLASHING are tasks which are managed by the PCS by itself.
- **Communication monitoring (wire-break, short circuit).** A very efficient data transfer is secured by the integrated priority management in connection with the intelligent package length optimization, the high thruput rate and the fault tolerance.

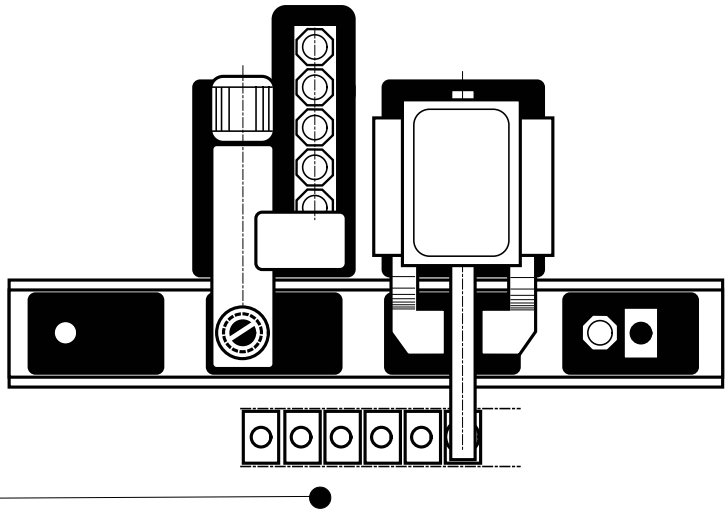
Operate and monitor

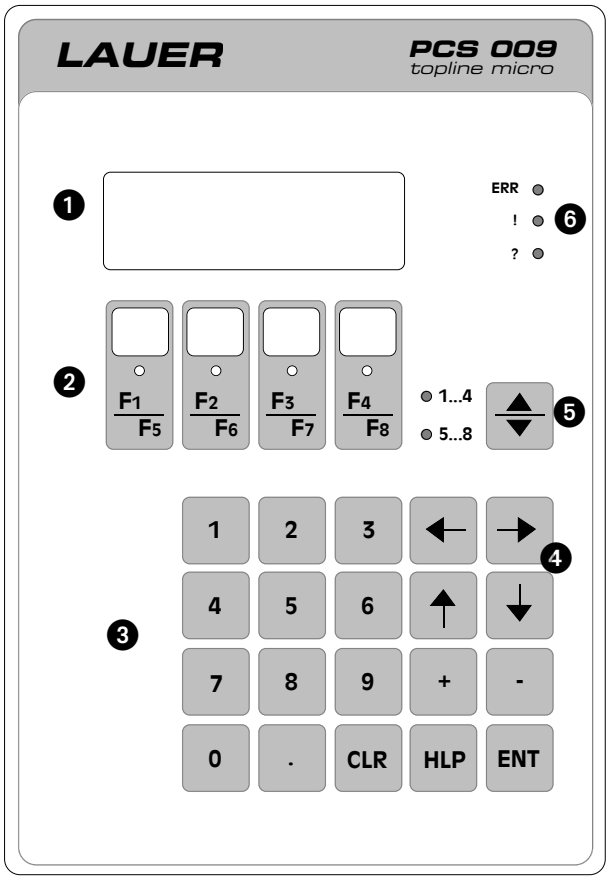


Control



Machine





Operator Panel PCS 009

❶ = LCD-Display, 4 lines each with 40 characters

❷ = Function keys (also as soft keys) F1...F8 with a greem message LED

❸ = 10 key keyboard for nominal value input

❹ = cursor and control keys for menus and nominal value input

❺ = Switch key (Shift key) for function keys (F1..F4, F5..F9)

❻ = Important information LEDs on the PCS status

❼ = Reset key

❽ = DIL switch for the PCS 009

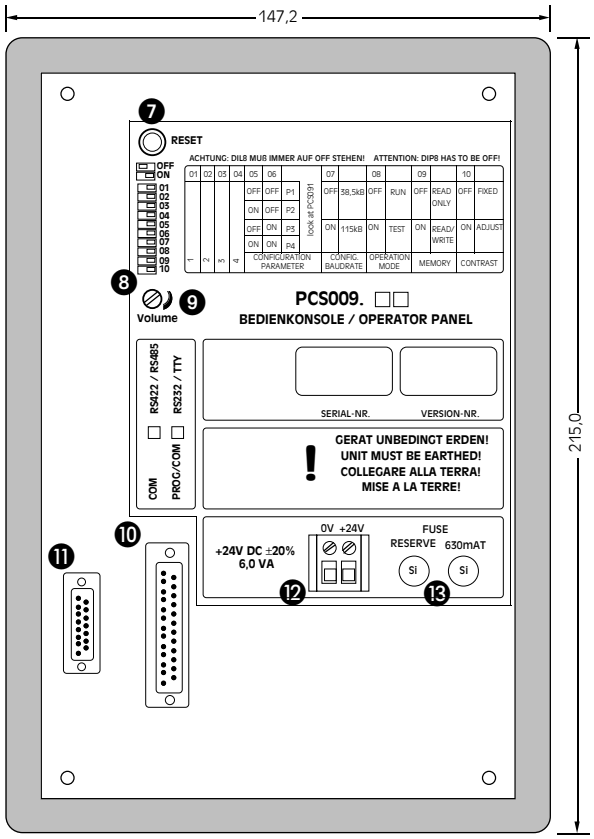
❾ = Volume for acoustic signal

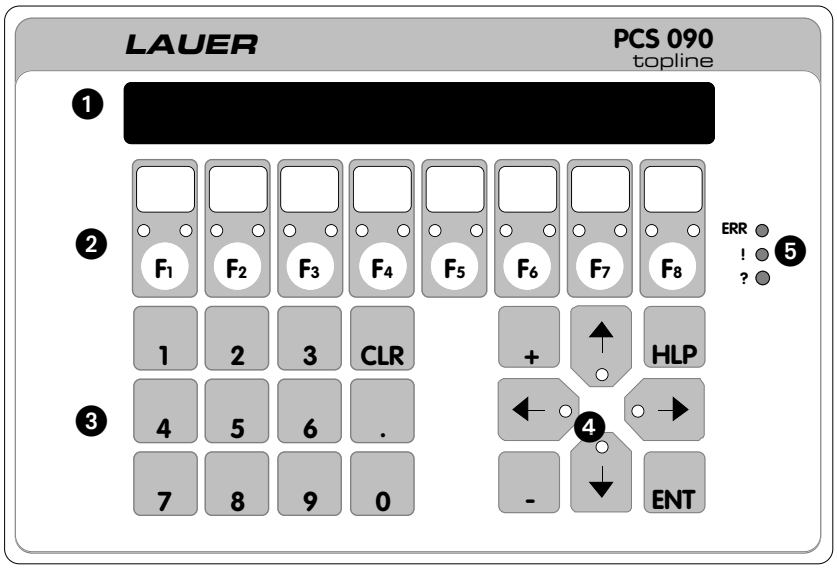
❿ = Serial interface RS 232/TTY for communication

⓫ = Serial interface RS 422/RS 485 for communication

⓬ = Operating voltage terminals

⓭ = Fuse with reserve fuse





Operator Panel PCS 090

❶ = LCD-display, 2 lines each with 40 characters

❷ = Function keys (also as soft keys) F1...F8 with a yellow and green message LED

❸ = 10 key keyboard for nominal value input

❹ = Cursor keys with LED and cursor control keys for menus and nominal value input

❺ = Important information LED's on the PCS status

❻ = Reset key

❼ = DIL-switch for the PCS 090

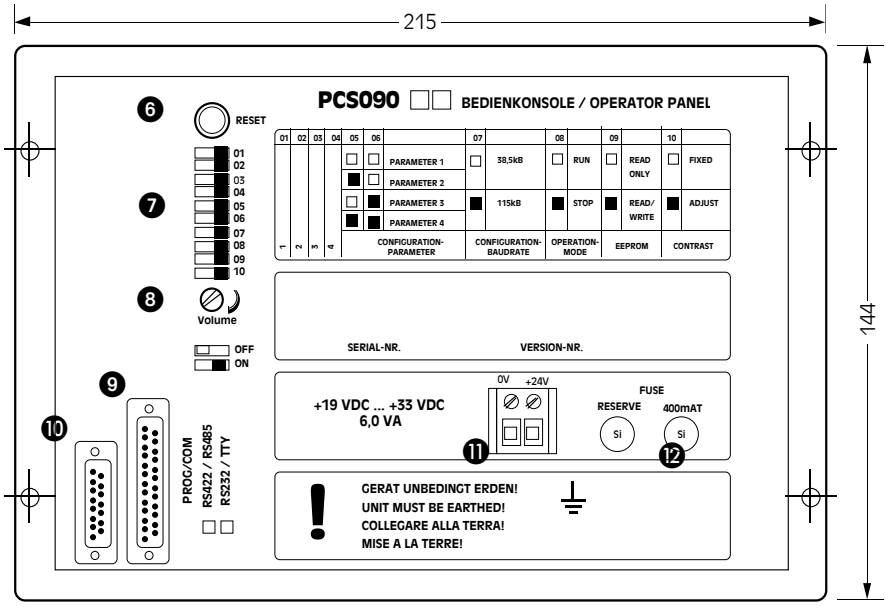
❽ = Volume for acoustic signal

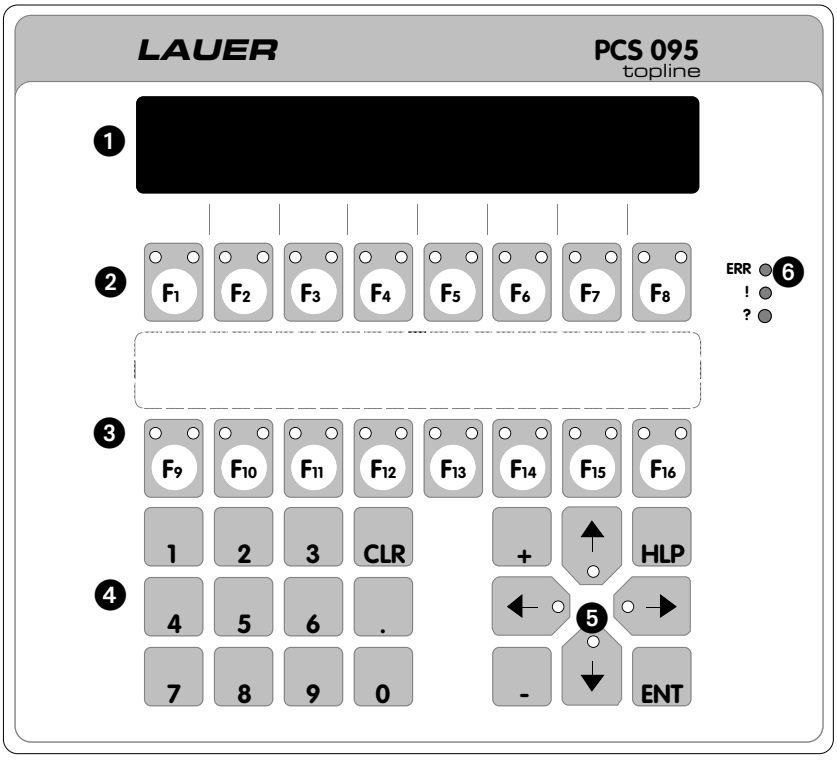
❾ = Serial interface RS 232/TTY for the communication

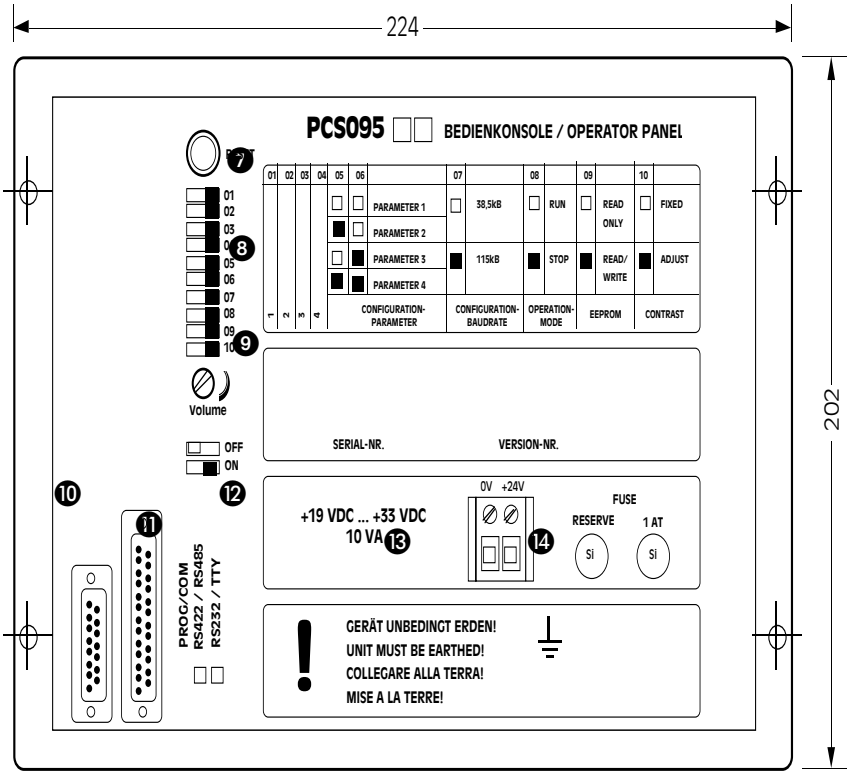
❿ = Serial interface RS 422/RS 485 for the communication

⓫ = Operating voltage terminals

⓬ = Fuse with reserve fuse







Operator Panel PCS 095/PCS 095.1

❶ = LCD-display, 4 lines each with 40 characters

❷ = Function keys (also soft keys) F1...F8 each with a yellow and a green message LED

❸ = Function keys F9...F16 each with a yellow and a green mesage LED

❹ = Ten key keyboard for nominal value input

❺ = Cursor key with LED and control key for menu and nominal value input

❻ = Important information LEDs on the PCS status

❼ = Reset key

❽ = DIL switch for the PCS 095

❾ = Volume for acoustic signals

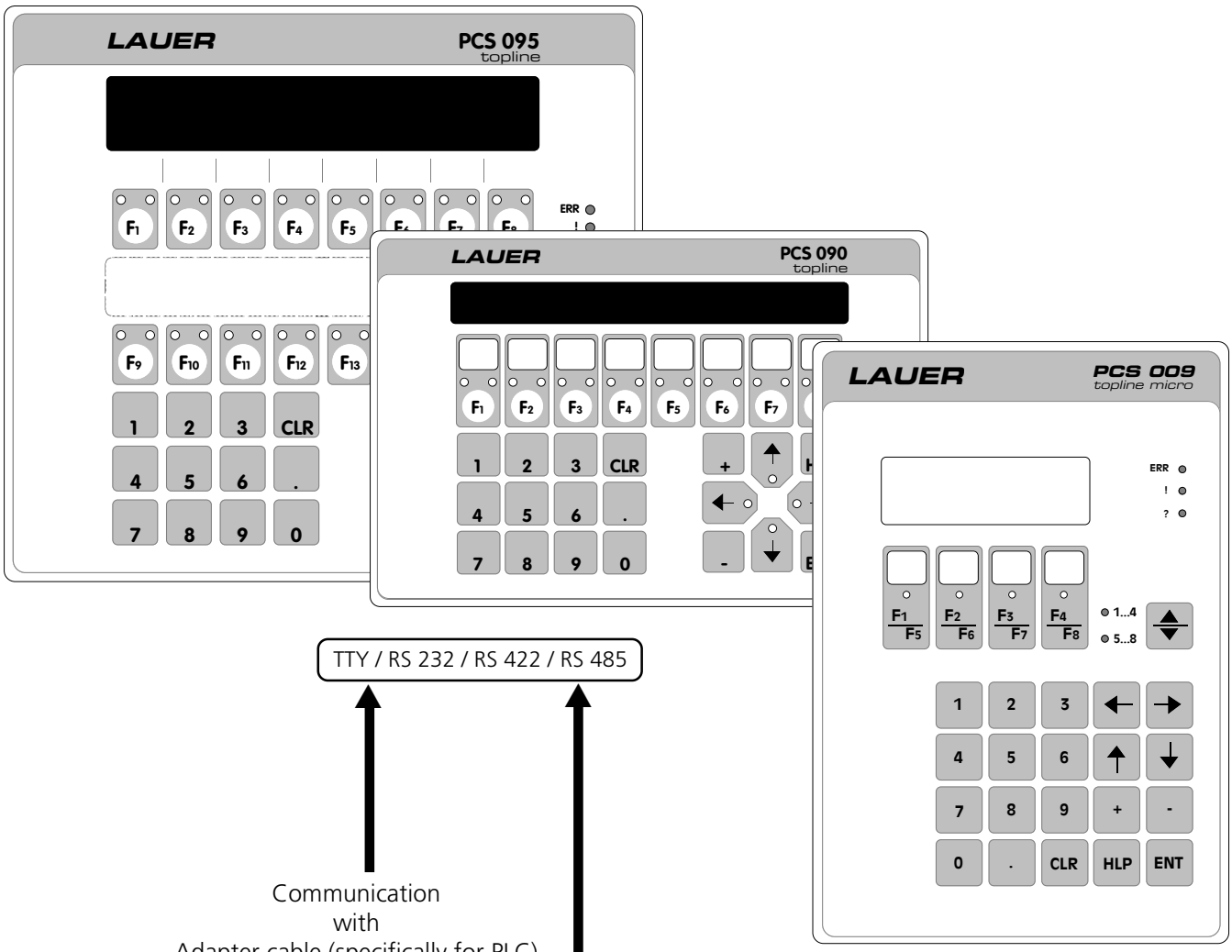
❿ = Serial interface RS 232/TTY for the communication

⓫ = Serial interface RS 422/RS 485 for the communication

⓬ = Serial interface RS 232/TTY for the programming and for the PCS 095.1 as Printer interface

⓭ = Operating voltage terminals

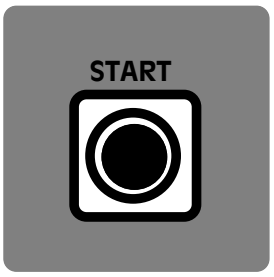
⓮ = Fuse with reserve fuse

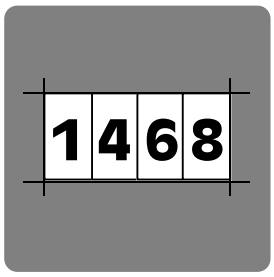


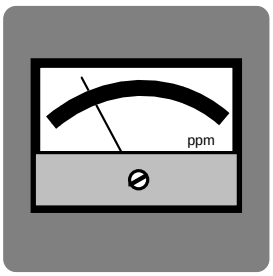
Programming with software
PCS*PRO*

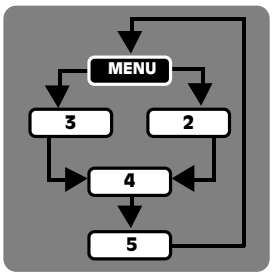
PCS is a universal operating concept for many PLC systems. The operating panels PCS 009, PCS 090, PCS 095, PCS 095.1 have a large selection of ready-made functions and tools to operate and monitor with:

8 function keys with 2 warning lights, red/green (OFF, ON, flashing)









650 switch with blank labelling for functions and switch setting

▲ Actual value digital indicator selection of 5 digits (0...65.535) or 10 digits (0...4.294.967.295)

128 message texts, 32 lines combined with 128 variables in 3 message priorities, with 3 message, 3 indicator and 4 deletion modes ▼

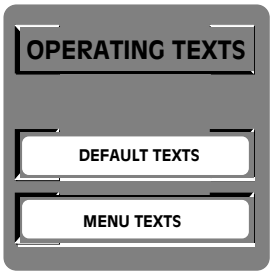
▲ Actual value analog indicator

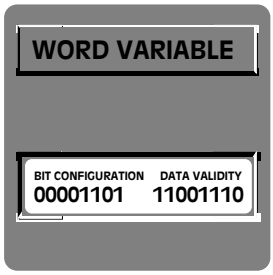
For menu and default texts there are 128, 2 line operating texts combined with 8 variables ▼

▲ 127 menues with max. 255 nodes or menu items. At PCS 090 each node is written into a 2 line menu text (with max. 8 variables) and at PCS 095, PCS 095.1 each node is written into a 4 line menu text (with max. 16 variables)

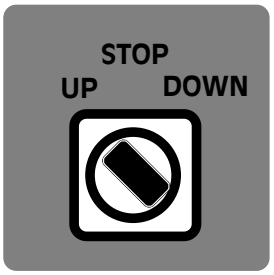
Notation and alteration of the bit configuration of a word in the PCS is possible at any time ▼

Keylock or codelock allocation for different gateways





233 selectors with up to 255 switch settings with blank labelling for functions and switch settings

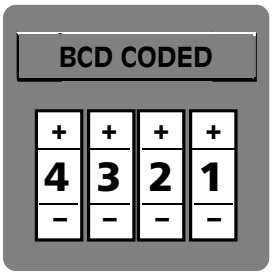


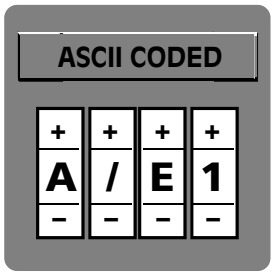
Digital ASCII nominal value input with ± key: 29 nominal value varables with 16 or 116 nominal value variables with 4 characters ▼

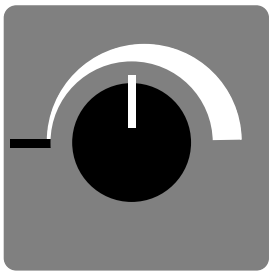
Analog nominal value input ▼

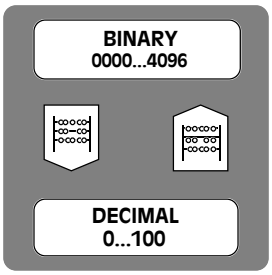
Automatic change of the nominal and actual values of the BCD/BIN in decimal and back with algebraic sign, limits and scale ▼

Digital BCD/BIN-nominal value input with ten key keyboard or ± key: 233 nominal value variables for 4 digit BCD or 5 digit BIN or 116 nominal value variables for 8 digit BCD or 10 digit BIN





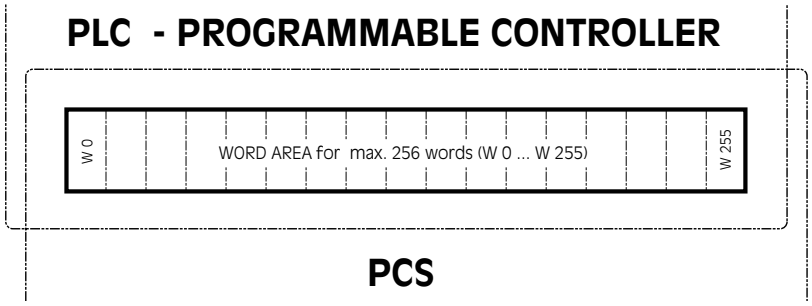




Communication between any PLC and the PCS occurs as follows:

The PCS writes in predetermined word areas of the PLC, functions or nominal values, which the PLC then reads and interprets.

The PLC writes in predetermined word areas functions or actual values, which are automatically read and interpreted by the PCS. Independently of the PLC there are maximum 256 words of 16 bit, that is to say 4096 inputs / outputs for the PCS/PLC communication available.



... and rapid set-up of a particular operational requirement

1 First define the specification and decide on the required PCS (PCS *micro*, PCS *mini*, PCS *midi* or PCS *maxi*)

2 Allocate the word and bit number to variables (actual- and nominal values).

3 Create the texts for operational guidance and help functions as well as for displays of machine conditions.

4 Determine the message texts and apply these words to them, subdivide the message texts into 3 priority groups

- Information
- Warnings
- Faults

and take into consideration the differing cancel modes, display and message modes. Display and message modes can be altered by the PLC at any time.

5 Define the menus and the menu operating texts.

6 Transfer the data file (variables, texts, menus) which was made in the PC or PG under MSDOS/DRDOS or compatible DOS-system, with the software PCS*PRO* into the PCS.

7 Implement and parameterize the PLC specific operating software (PCS 91.nn, see overall view of information) in the users' programme.

8 Connect the PCS via the adapter cable with the PLC. Test together the operation and control of the PCS and PLC and adjust if necessary.

Machines produce different parts. Therefore quick and selective alterations of finished sizes and functions (variables) are especially important for increased flexibility.

The PCS features a convenient method of processing the variables. 650 external variables (freely definable) and 6 internal variables are supervised from the PCS.

The value of the external variables are stoped in the words 30...255. The PCS differentiates between actual values and nominal values:

ACTUAL: The value in the word is the actual value. The PCS can only display the value.

NOMINAL: The standing value in the word is the nominal value. The value can be displayed and changed by the PCS.

NOMINAL VALUE-P: The private value in the word is a nominal value. The PCS can display the value. It can be changed only if this is allowed by the word 14 bit 7 = log 1 (key switch or DIL-switch 1...4 on the rear side of the PCS). When the bit 7 of word 14 = log 0, the display of the actual value follows.

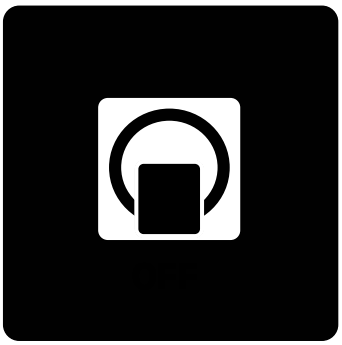
Internal Variables

NAME	CONTENTS	FORMAT	LENGTH	ACTUAL/NOMINAL
ZP	NUMBER OF INFORMATIONS	BIN	3	ACTUAL
ZQ	NUMBER OF WARNINGS	BIN	3	ACTUAL
ZR	NUMBER OF FAULTS	BIN	3	ACTUAL
ZT	MENU NUMBER	BIN	2	ACTUAL
ZV	SCROLL TIME	BIN	2	NOMINAL
ZX	INTERFACE FAULTS	BIN	2	ACTUAL

External Variables

FORMAT,	LENGTH
BIT variable	max. length 40 characters
STRING variable	max. length 40 characters
CSTRING variable	max. length 40 characters
WORD variable KM, KH, KY:	length: 17, 4, 7 characters
ASCII variable	max. length 16 characters
BCD-1 variable	max. length 4 digits
BCD0-1 variable *)	max. length 4 digits
BCD-2 variable	max. length 8 digits
BCD0-2 variable *)	max. length 8 digits
BIN-1, BIN-A variable	max. length 16 bits/11 digits
BIN0-1, BIN0-A variable *)	max. length 16 bits/11 digits
BIN-2, BIN-B variable	max. length 32 bits/11 digits
BIN0-2, BIN0-B variable *)	max. length 32 bits/11 digits
VBIN-1, VBIN-A variable	max. length 16 bits/11 digits + operational sign
VBIN0-1, VBIN0-A variable *)	max. length 16 bits/11 digits + operational sign
VBIN-2,V BIN-B variable	max. length 32 bits/11 digits + operational sign
VBIN0-2, VBIN0-B variable *)	max. length 32 bits/11 digits + operational sign
Timer variable	max. length 40 characters

*) BIN0...- and VBIN0... variable are only programmable with PCS*PRO*



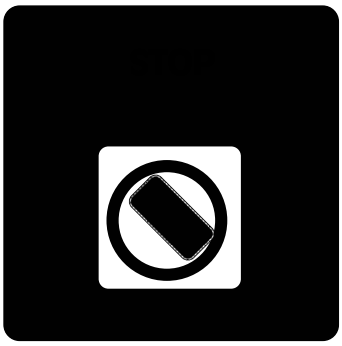
The BIT variable

When two possibilities can be selected at an input, the descision is taken by the bit variable. This is in the form of an ON/OFF switch.

Every switch position represents an inscription (text) which appears in the display. Each bit variable occupies a bit. A data word can also take on up to 16 differing bit variables or switches.

Example: A wood shavings vacuum absorption cleaner shall be switched on or off in bit 2 data word 33.

The +/- switch selects the inscription or the switching position. The bit bears the value of the inscription. The first inscription carries the value log 0, the second bears the value log 1.



STRING variable

When two or more possibilities can be selected at an input, the decision is made by the STRING variable. It corresponds to a selector switch.

With STRING variables, every switch position is classified with an inscribed text, which appears in the display. Every STRING value carries a data word with up to 256 switch positions. The switch position is deposited in low bytes of the data word.

Example: The frame material shall be selected in data word 40.

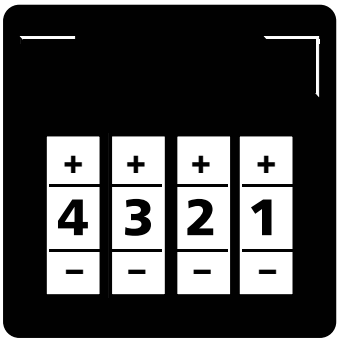
The inscription or switching position is selected with the +/- key. Acceptance follows with the ENTER key.

CSTRING variable

The CSTRING variable corresponds to STRING variable. Acceptance follows directly after using the +/- key without ENTER.

BCD variable: BCD-1, BCD-2, BCD0-1, BCD0-2

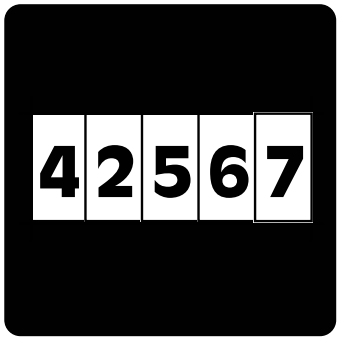
The nominal value of the BCD variable corresponds to that of a BCD thumbwheel switch and the actual value to that of a BCD digital display. The 4 digit (decimal positions) variable BCD-1 is allocated to a word, the 8 digit variable BCD 2 to two consecutive following words 32 bits (W n, W n+1). BCD variables are displayed without pre-zeros. For example a BCD 2 actual value is: 4 2567.



The 4 digit variable BCD0-1 is allocated to a word, the 8 digit variable BCD0 to two consecutive following words 32 bits (W n, W n+1). BCD0 variables are displayed with pre-zeros. For example the actual value of a BCD0 is: 0004 2567

Every BCD value is limited to a min/max value.

Example: The batch size per window type is written in word 30 as a 4 digit nominal value without min/max limits.



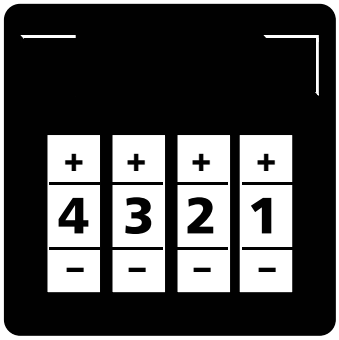
The nominal value input "8500" takes place with the ten key keyboard of the PCS and is transfered to the word with the ENTER BCD coded key:

Word 30 = 1000 0101 0000 0000

8 5 0 0

BINARY variables: BIN-1, BIN-2, BIN-A, BIN-B, VBIN-1, VBIN-2, VBIN-A, VBIN-B, BIN0-1, BIN0-2, BIN0-A, BIN0-B, VBIN0-1, VBIN0-2, VBIN0-A, VBIN0-B

The nominal value of the BIN variable corresponds to that of a BIN thumbwheel switch and the actual value to that of a BIN digital display. The scaled 16 bit variables (BIN-1 to VBIN-A) are allocated to a word, the 32 bit variables (BIN-2 to VBIN-B) to two consecutive following words (W n, W n+1).

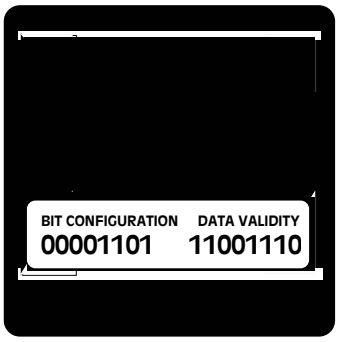


The variables (V)BIN(0)-1,2 only differ from those of the (V)BIN(0)-A, B in the way the characters are loaded (V)BIN0-1,2 and (V)BIN-1,2 are with and without pre-zeros repectively VBIN(0)-1,2A,B take the operational sign into consideration. Every (V)BIN variable is limited by a min/max value. In addition the (V)BIN-1,A variables can be scaled.

Example: The temperature of the engine brake can be set between 0°C and 70°C. The scaled 16 bit nominal value is written into the word W 45 with a min/max limit.

Example: The window height is adjustable between 750 mm and 1500 mm. The 32 bit nominal value is written into the word 41+42 with a min/max limit. The higher value part remains in W 41, the lower value part in W 42.

WORD variable



The WORD variable is specially suitable for service. Die Darstellung kann sowohl bitweise (KM), hexadezimal (KH) oder byteweise dezimal (KY) erfolgen. An alteration of the bit pattern with the PCS is possible if the WORD variable is defined as a nominal value.

Example: The word 33 is to be displayed and altered in the PCS display:

Alteration of the nominal value WORD variable takes place in a menu.

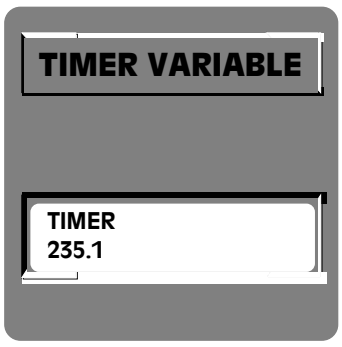
Display line 1 *BIT PATTERN OF WORD 33*
Display line 2 *00000000 00000101*

The value of the WORD variable can be changed with the "0" or "1" key. The "+" key switches the pointer one place to the right and with the "–" key, one place to the left.

Display line 1 *BIT PATTERN OF WORD 33*
Display- line 2 *11110000 11000000*

The ENTER key puts the new value into the word.

TIMER variable



The TIMER variable allows an input of a 3 digit numeric input (BCD) and a timebase value with 4 selection (displayed as text).

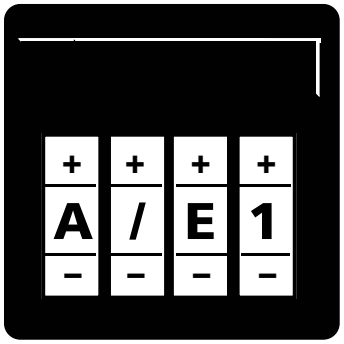
Example: Word 100 should be displayed in Timerformat.
The content of word 100 is KH1235.
If the timebase is defined as ".0", ".1", ".2" and ".3" in the display "235.1" is visible.

The value is defined with the key "0"..."9". The selection between base and value is done with the (.) key. The timebase is selectable with + or - key or direct input with "0" bis "3" key.

ASCII variable

If an alphabetical nominal value is required (article number, name etc.) the ASCII variable provides it in a simple manner.

Example: Enter the 12 digit version » 41-BN-890-SB «:



As every 2 ASCII characters occupy a word, 6 words are to be reserved for a 12 digit version number. In the following example the words 56...61 are used to this purpose.

The loading of the nominal value ASCII variable takes place in a menu.

Display line 1 *LOADING OF VERSION:*
Display line 2 ■■■■■■■■■■■■

By calling the menu the value 0 stands in the DW 56 ... DW 61. For this value the PCS-ASCII chart sets up the signs "■" (all dots illuminate). By using any key these signs will be replaced by a question mark (?).

Display line 1 *LOADING OF VERSION:*
Display line 2 *???????????*

Every "?" can be changed with a "+" key to any letter desired and with the "-" key to any character required. The "point" key moves the indicator one place to the right.

Display line 1 *LOADING OF VERSION:*
Display line 2 *41-BN-890-SB*

When all characters have been completely and correctly loaded, pressing the ENTER key for example, puts the values into the words 56 ... 61. The words then have the following values:

Word No.	*Contents ($)*	*ASCII Characters*
W56	*34 31*	*4 1*
W57	*2D 42*	*- B*
W58	*4E 2D*	*N -*
W59	*38 39*	*8 9*
W60	*30 2D*	*0 -*
W61	*53 42*	*S B*

The keyboard outline of the ASCII variables

+ key	pages to the letters (characters with the next largest ASCII code)
- key	pages to the characters (characters with the next smallest ASCII code)
Point key	moves the cursor to the right
ENTER key	records the ASCII characters into the data words
CLR key	displays old value

The number and format of nominal values are as varied as the operation itself. Regardless of the type and number of nominal values required, the procedure for recording them used by "the man at the machine" must be simple and straight forward.

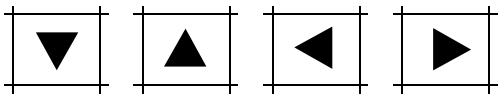
The menu technique offers considerable flexibility in recording and altering nominal values. It guides the operator and eliminates almost any possibility of false entering.

The PCS has at its disposal:

- 127 menus with a maximum of 255 menu or node points

Every node can be written with a 2 (PCS 090) or 4 (PCS 009, PCS 095, PCS 095.1) line operator text. This text can contain a maximum of 8 (PCS 009, PCS 090) or 16 (PCS 095, PCS 095.1) variables (nominal values/actual values).

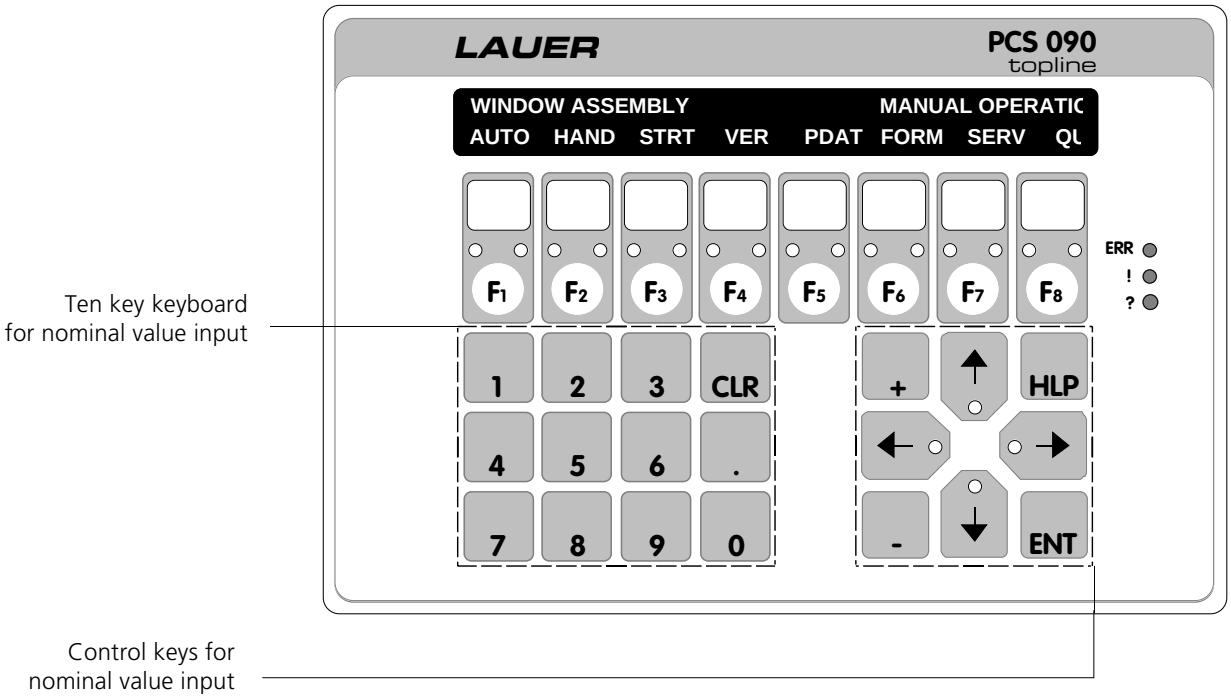
The PLC calls a menu with the word W 14 (bit 0...6). The PCS always shows the text of the start node. Depending on the arrangement of the menu, the other nodes are reached by actuating the ARROW key. The LED in the arrow key shows the operator the direction in which further variables (nominal values) are to be edited., i.e., the relevent LED lights. If on the other hand an LED flashes, it shows the operator that this node will be left on activating. Termination of a menu is achieved by setting bit 0...6 back in word 14.



The PCS has a simple editor for entering functions and nominal values. This editor permits 3 different inputs of figures:

- Nominal value input with the numeric key pad
- Incrementing/Decrementing the nominal value with the +/- key
- Addition and subtraction of various values of the displayed nominal value (only with BCD and BIN variables)

The CLR key sets nominal value back to its old value.



Operation of the integrated editors		
Variable Type	Key	Function
BIT	PLUS	A bit that was logic 0 sets to logic 1 (at once written in the PLC).
	MINUS	Deletes a bit that was logic 1 to logic 0 (at once written in the PLC).
	* ARROWS	Leaves this variable if allowed. The next variable or node in arrow direction is looked for.
STRING	* PLUS	Increments the value of a variable, so long as value is still within limiting values.
	* MINUS	Decrements the value of a variable, as long as value is still within limiting values.
	CLR	Restore the old value in the display; (the last value read by the PLC).
	ENTER	Writes the selected value in the PLC as long as it has been amended and not yet written.
	* ARROWS	Write the selected value, if it has been modified and not yet sent and then look for the next variables, i.e., the next menu nodes in direction of arrow.
CSTRING	* PLUS	Increments the value of a variable, so long as the value is still within the limiting value (in contrast to STRING at once stored in the PLC).
	* MINUS	Decrements the value of a variable, so long as the value is still within the limiting value (in contrast to STRING at once stored in the PLC).
	CLR	Restore the old value into the display; (the last read value of the PLC).
	* ARROWS	Leave these variables if allowed to. The next variable or node in arrow direction looked for.

* = Auto repeat

Variable Type	Key	Function
BCD-1 BCD-2 BCD0-1 BCD0-2	* PLUS/ MINUS	Adds/subtracts n within the limiting values (Offset input), whereby * n = 1 if no numeral input follows, i.e., * n = given value, if numerical input follows.
	CLR	Restores old value in the display; (last read value by PLC)
	ENTER	Writes the selected value in the PLC, if it has been changed and not yet written.
	* ARROWS	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next node in the arrow direction.
	* NUMBERS	Permit direct input.
BIN-A BIN-B BIN0-A BIN0-B	* PLUS/ MINUS	Adds/subtracts n within the limiting values (Offset input),whereby * n = 1 if no numeral input has occured, i.e., * n = value input, when numerical input has just been entered.
	CLR	Restores the old value in the display; (last value read by PLC).
	ENTER	Writes the selected value in the PLC, if it has been changed and not yet written.
	* ARROWS	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction.
	* NUMBERS	Enables direct input. Numbers moved from right to left (even beyond a decimal point).
BIN-1 BIN-2 BIN0-1 BIN0-2	* PLUS/ MINUS	Adds/subtracts n within the limiting values (Offset input), whereby * n = 1 if no numerical input follows, i.e., * n = given value, if numerical input follows.
	CLR	Restores the old value in the display; (last value read by PLC). ENTERWrites the selected value in the PLC, if it has been changed and not yet written.
	* ARROWS	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction.
	* NUMBERS	Enables direct input. Numbers are entered according to the pocket calculator principle.
	(*)POINT	Changes to after decimal point position, if after decimal point positions are defined.

* = Auto repeat; (*) = Auto repeat, though without significant meaning

Control keys and there functions in the nominal value input

Variable Type	Key	Function
VBIN-A VBIN-B VBIN0-A VBIN0-B	* PLUS * MINUS CLR ENTER * ARROWS * NUMBERS	Gives the operational sign "+". Gives the operational sign "-". Restores old value in the display; (last read value by PLC). Writes the selected value in the PLC, if it has been changed and not yet written. Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next node in the arrow direction. Permit direct input. Numbers moved from right to left (even beyond a decimal point).
VBIN-1 VBIN-2 VBIN0-1 VBIN0-2	* PLUS * MINUS CLR ENTER * ARROWS * NUMBERS (*)POINT	Gives the operational sign "+". Gives the operational sign "-". Restores the old value in the display; (last value read by PLC). Writes the selected value in the PLC, if it has been changed and not yet written. Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next node in arrow direction. Enables direct input. Numbers entered according to the pocket calculator principle. Changes to after decimal point positions, if after decimal point positions are defined.
WORD	* PLUS * MINUS CLR ENTER * ARROWS * NUMBERS	Moves the cursor one bit position to the right in direction of lowest value bit LSB. Moves the cursor one bit position to the left in the direction of the highest bit MSB. Restores the old value in the display; (last value read by PLC). Writes the selected value in the PLC, if it has been changed and not yet written. Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction. Only the keys <0> and <1> are significant: <0> Sets a bit to 0 and moves the cursor, if possible one position tho the right. If the cursor finds itself at end of the variables, then it will be positioned on the highest bit (MSB). <1> Sets a bit to 1 and moves the cursor, if possible one position to the right. If the cursor finds itself at end of the variables, then it will be positioned on the highest bit (MSB).

* = Auto repeat; (*) = Auto repeat, though without significance

Variable Type	Key	Function
ASCII	* PLUS	Presents the character with the next higher displayable character code. If the end of the character table has been reached, the first presentable character is returned.
	* MINUS	Present the character with the next smaller displayable character code. If the begining of the character table has been reached, the last character out of the character table is returned.
	CLR	Restores the old value in the display; (last value read by PLC).
	ENTER	Writes the selected value in the PLC, if it has been changed and not yet written.
	* ARROWS	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction.
	* POINT	Moves the cursor one place to the right. If the variable end has been reached, then the cursor will again be set on the first position of the variables.
	* NUMBERS	Used for direct entry: Numbers are shifted from right to left (calculator entry).

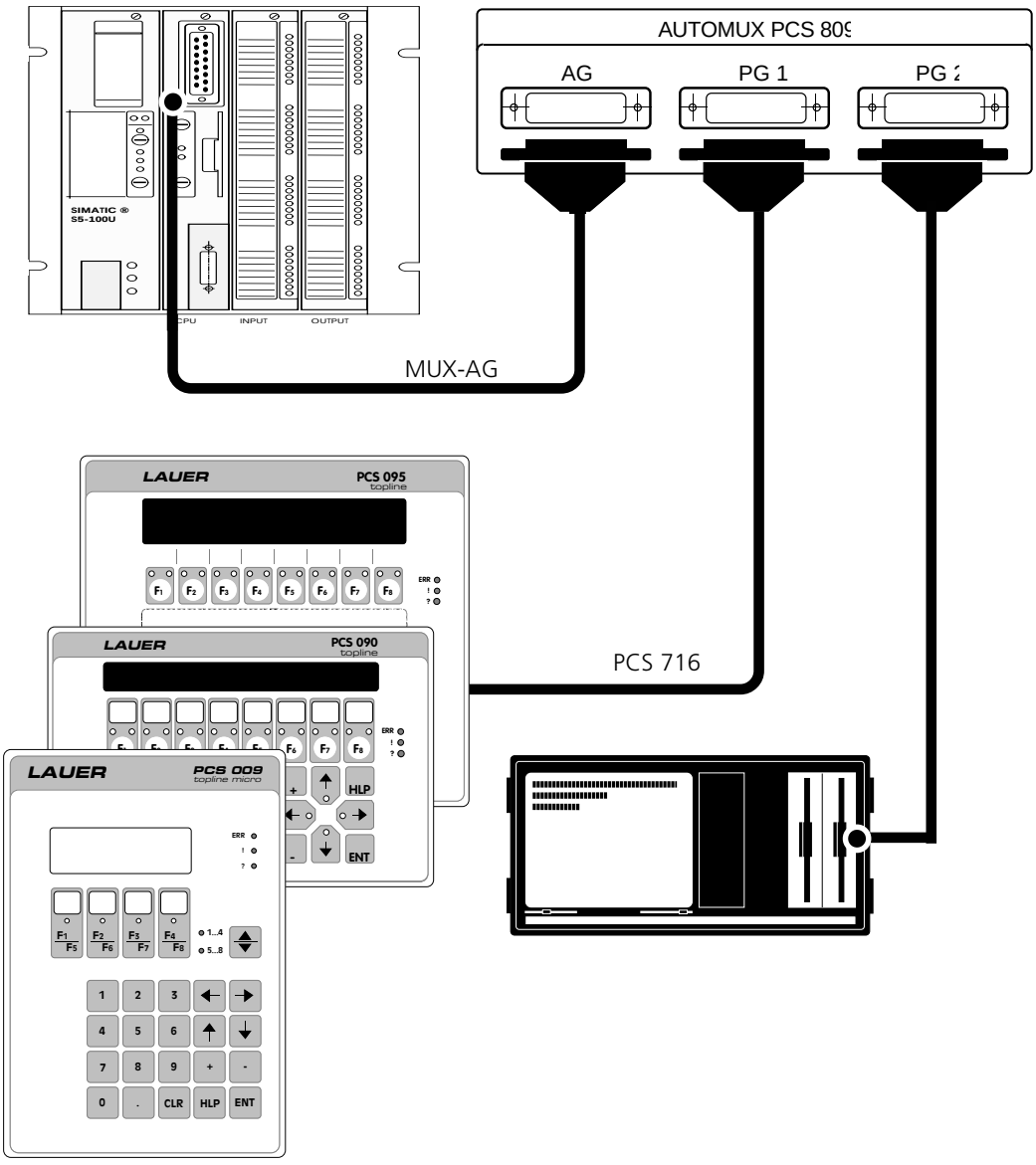
* = Auto repeat

The characters can be presented on the LCD display. 8 characters are individually definable.

00	10	20	0 30	@ 40	P 50	\ 60	p 70
◀ freely defined characters		! 21	1 31	A 41	Q 51	a 61	q 71
		" 22	2 32	B 42	R 52	b 62	r 72
		# 23	3 33	C 43	S 53	c 63	s 73
		$ 24	4 34	D 44	T 54	d 64	t 74
		% 25	5 35	E 45	U 55	e 65	u 75
		& 26	6 36	F 46	V 56	f 66	v 76
		´ 27	7 37	G 47	W 57	g 67	w 77
08		(28	8 38	H 48	X 58	h 68	x 78
09		) 29	9 39	I 49	Y 59	i 69	y 79
0A		* 2A	: 3A	J 4A	Z 5A	j 6A	z 7A
0B		+ 2B	; 3B	K 4B	[5B	k 6B	{ 7B
0C		, 2C	< 3C	L 4C	¥ 5C	l 6C	| 7C
0D		— 2D	= 3D	M 4D	] 5D	m 6D	} 7D
0E		. 2E	> 3E	N 4E	^ 5E	n 6E	-> 7E
0F	1F	/ 2F	? 3F	O 4F	— 5F	o 6F	<- 7F

The PG interface is occupied if the communication between the PCS and the Siemens S5 runs via the L1 standard or L1 direct protocol.

As the smaller PLC systems only have a PG interface, this leads to problems during running as a simultaneous application of PG and PCS is not possible.



The Automux PCS 809 removes this problem. The PCS 809 broadens the PLC-PG-interface so that the PG and the PCS can serve the PLC together. The switch over to MUX follows automatically.

The PCS 809 is intended to be used as a commissioning tool. After the start-up procedure the PCS is connected to the PLC via the PG interface. We recommend the Automux PCS 809 for the Siemens PLC

S5-90U
S5-95U
S5-100U
S5-115U (CPU with one interface)

PCS 809 is valid for the PCS Operator Panel

PCS 009
PCS 090
PCS 095/095.1
PCS 900
PCS 950
PCS 950C
PCS 9000
PCS 110
PCS 210

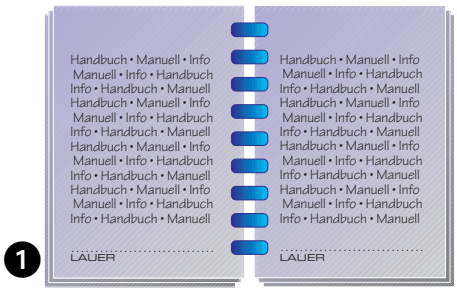
We supply the PCS 809 with power supply cable and adaptor cable MUX / PLC-AG.

Technical Manual
PCS 091

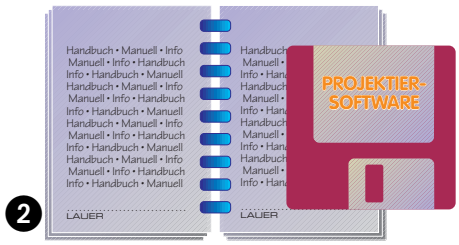
for the operating consoles
PCS 009, PCS 090, PCS 095, PCS 095.1

TABLE OF CONTENTS

MANUAL ORGANIZATION FOR THE OPERATING CONSOLES AND PROGRAMMABLE CONTROLLER DRIVERS



❶ Manual PCS 091
for the operating consoles
PCS 009, PCS 090,
PCS 095, PCS 095.1



❷ Introduction PCS*PRO*
Setup software for the
PCS 009, PCS 090,
PCS 095, PCS 095.1

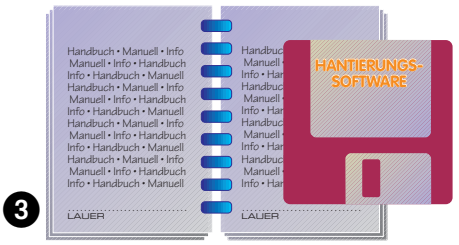
You need the PCS 091 Technical Manual for the operating consoles PCS 009, PCS 090, PCS 095, PCS 095.1.(❶

For the setup of the operating consoles you need the setup software PCS*PRO*. We supply the software with a brief introduction. The extensive help system of PCS*PRO* supports you directly on-screen (❷

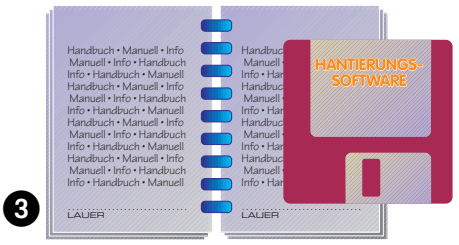
Use the appropriate driver for easy communication of the PCS with your programmable controller. As an appendix to the PCS 091 manual you receive a detailed driver description with the handling module which is delivered on a 3.5" floppy disk (❸ The following order numbers apply to the various driver appendixes*):

PCS 91.ABB	for ABB progr. contrl.
PCS 91.AEG	for AEG progr. contrl.
PCS 91.ALB	for Allen-Bradley progr. contrl.
PCS 91.B+R	for Bernecker & Rainer progr. contrl.
PCS 91.BOS	for Bosch progr. contrl.
PCS 91.CEG	for Cegelec progr. contrl.
PCS 91.CRO	for Crouzet progr. contrl.
PCS 91.EBE	for Eberle progr. contrl.
PCS 91.FES	for Festo progr. contrl.
PCS 91.GEF	for GE-Fanuc progr. contrl.
PCS 91.HIT	for Hitachi progr. contrl.
PCS 91.IPC	for IPC progr. contrl.
PCS 91.IZU	for Izumi/Idec progr. contrl.
PCS 91.KLM	for Kloeckner-Moeller progr. contrl.
PCS 91.MAT	for Matsushita progr. contrl.
PCS 91.MIT	for Mitsubishi progr. contrl.
PCS 91.OMR	for Omron progr. contrl.
PCS 91.PHI	for Philips progr. contrl.
PCS 91.SAI	for Saia progr. contrl.
PCS 91.SEL	for Selectron progr. contrl.
PCS 91.SIE	for Siemens progr. contrl.
PCS 91.TEC	for Teco progr. contrl.
PCS 91.TMQ	for Telemecanique progr. contrl.
PCS 91.TOS	for Toshiba progr. contrl.

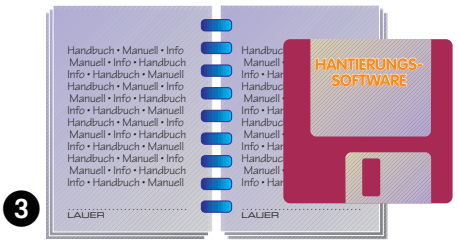
*) Driver status 6/95



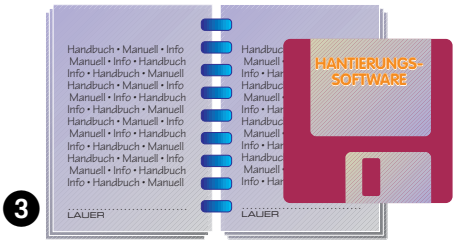
❸ Appendix to the PCS 091
Handling software
PCS 91.AEG
AEG



❸ Appendix to the PCS 091
Handling software
PCS 91.GEF
GE-Fanuc



❸ Appendix to the PCS 091
Handling software
PCS 91.SIE
Siemens



❸ Appendix to the PCS 091
Handling software
PCS 91.KLM
Klöckner-Moeller

Please take care to use the appropriate adapter cable for the programmable controller

and more appendixes

0 IMPORTANT USER NOTES

0.1 IDEOGRAMS AND SYMBOLS

The following symbols and ideograms are used in this manual.

Warning!
Possibly dangerous situation which can cause death and most serious injuries.

Caution!
Possibly dangerous situation which can cause light and less serious injuries.

Attention!
Possibly harmful situation which can cause damage to the product or its environment.

Mechanical pressure causes damage to the product.

Information concerning safety when using the devices In an ex area.

Information and notes which must additionally be observed.

0.2 SAFETY RELATED INFORMATION

- The device may only be connected to the systems specified by Systeme Lauer.
- The device meets the current technical state of the art.
- Only trained and qualified persons who have familiarized themselves with the product are allowed to install and operate the device.
- The responsibility of persons operating the device must be clearly determined in order to avoid undefined competencies.
- The relevant safety regulations and standards must be observed.
- Opening the device is not allowed. Systeme Lauer is not responsible for resulting damages.
- Before commissioning the device, this instruction manual must be read thoroughly.
- Modifications of or changes to the design of the device are not allowed. Systeme Lauer is not responsible for resulting damages.
- The supply voltage of the device must be within the range specified in the section „Specifications". Systeme Lauer is not responsible for damages resulting from non-compliance to this requirement.
- The latest manuals and documentations are valid.

The specifications published by Systeme Lauer were determined with our methods and facilities; characteristics are only guaranteed in this respect. The user is responsible for testing and determining the suitability for the specific application or for use under actual conditions. Systeme Lauer does not assume any warranty for this.

Modifications reserved

1 GENERAL REFERENCES

1.1 BREAKDOWN OF THE MANUAL

The manuals are divided up similarly to the structures of PCS 009, PCS 090, PCS 095, PCS 095.1.

The PCS possess an EPROM area in which part of the internal hardware of the client are to be found. Alongside this there is still a programmable EEPROM-storage area (zero dielectric strength), in which there is the data record (variables, texts and menus), as well as the driver for communication with an PLC and an optional additional function. The PCS can send out no communications unless there is a program in this memory. The PCS is supplied with example texts and a driver in the storage.

The operation and loading of the data record, as well as the driver, are described in a separate manual PCS*PRO* . They are the same for all loadable drivers. The corresponding programming software for an IBM compatible PC or PG is described in these manuals.

Warning!
Creation of a data record is only possible by means of the PCS*PRO* software. Other software packages are inadmissible and may cause malfunctions in the PCS and in the programmable controller.

Information about the special drivers, as well as the adaption of the PCS to an PLC system are described in separate manuals PCS 91.xxx (for example PCS 91.SIE, Coupling PCS with Siemens by means of the "LAUER" driver). These manuals differ according to the driver and connected up PLC system. Disks of the required operating software for the PLC are enclosed with these manuals.

Warning!
Use only the drivers specified for the programmable controller. Other drivers may cause malfunctions in the PCS and in the programmable controller.

The data record defines which data, that is to say words are to be applied in the actual PLC and as to how the PCS should react to altered data.

The driver book PCS 91.xxx describes the procedures and the components which are required in order to enable a PCS to communicate with an PLC system.

It must be defined in the PLC programme how the actual PLC is to react to data and as to how it should also present data.

1.2 GENERAL PROCEDURES

Please follow the description below to setup a complete system:

- Specify the functions of the system.
- Create a data record with the required parameters (variables, texts, menus) and download it into the PCS 009, PCS 090, PCS 095, PCS 095.1 using your specific driver. Refer to the PCS*PRO* manual and to this part of the manual for more information.
- Write a programmable controller program (information is contained in the driver appendix PCS 91.xxx) and download it into the system.
- Connect the PCS with the programmable controller. Test the communications and solve any faults.

1.3 TEXT ASSIGNMENTS

The information on the following pages refer to the PCS 009, PCS 090, PCS 095, PCS 095.1. In order to create the operating programm, i.e., configuration of the PCS, it is necessary to use the PCS*PRO* manual.

The communication of the PCS takes place via a word range "transfer range" (word 0 ... max. words 255) and can be parametrized according to the PLC system be currently used. The words are numbered with W0 ... W255 for use in this manual. Information regarding the actual PLC system in use can be found in the appropriate driver manual PCS 91.xxx.

The following abbreviations symbols are used in the manual:

$ is an abbreviation for the hexadecimal presentation of data

[+] represents a key on the PCS 009, PCS 090, PCS 095, PCS 095.1 and here specifically the plus key.

1.4 EQUIPMENT AND ACCESSORIES REQUIRED

To write a user program and transfer this program into the PCS together with a driver. The following (Systeme Lauer) products are required:

1. The PCS itself
2. The programming cable PCS 733 for programming the PCS using an IBM compatible PC or programmer.
3. This manual (PCS 091).
4. The PCS*PRO* programming manual with diskette.
5. Driver manual (PCS 91.xxx, depending on the driver required).
6. For "beginners" we recommend the "PCS-SKILLS" booklet with an example program for the PCS.

The following are also required:

7. An IBM compatible PC or programmer with MSDOS > 3.3 or DRDOS operating system and at least one serial interface (COM).

... also the power supplies for all components.

2 FUNCTION

2.1 FUNCTIONAL OVERVIEW

- **Machine operation using 8 (PCS 009, PCS 090) or 16 (PCS 095, PCS 095.1) freely assignable keys** These F01 to F08/F16 labeled keys can be application specifically inscribed and are provided to the controller as status bits.
- **Machine operation using 4 (PCS 009), 16 (PCS 090) or 32 (PCS 095, PCS 095.1) freely usable LEDs** These can be assigned the indicating states ON, DARK, FLASHING, and INVERSE FLASHING. A green and a yellow LED is allocated to each function key.
- **Display of fixed texts with integrated variable values** The values can be represented selectably as numerical values or in text format.
- **Representation of the contents of 233 words as variables** In addition, 650 external variables can be defined. **9 variable formats** (from bit to timer) are available.
- **3 text groups** 128 operating texts as menu and idle texts, 128 message texts with up to 332 lines, 5 help texts with up to 32 lines.
- **127 menus with 255 menu nodes** each for any menu configurations.
- **4 different deletion modes** For every message, 1 of 4 possible deletion modes can be selected.
- **Modification of the content of any word within the transfer area** Using the integrated editor all possible representation formats can be setup.
- **7 priority levels** For idle text up to help text, 3 message priorities Information, Warning, Fault. This working-condition related management significantly off-loads the programmable controller program.
- **Monitoring of rising or falling edges of 128 consecutive bits** The assignment of texts, the management of 3 priority levels (Information, Warnings, and Faults), keeping the timely sequence as much as possible, organization of the FIRST MESSAGE, LAST MESSAGE, and CYCLIC DISPLAY, the individually settable deletion behaviour, and the representation formats NORMAL and FLASHING are tasks which are managed by the PCS by itself.
- **Communication monitoring (wire-break, short circuit)** A very efficient data transfer is secured by the integrated priority management in connection with the intelligent package length optimization, the high thruput rate and the fault tolerance.

2.2 OPERATING ELEMENTS

2.2.1 DIL SWITCH

On the rear side there are 10 DIL switches.

DIL 1 to 4	=	PLC bits. These switches are in word 4, bit 4 to 7 are freely available DIL 1 = W4.4 DIL 2 = W4.5 DIL 3 = W4.6 DIL 4 = W4.7
DIL 5, DIL 6	=	Configurations parameter (driver) e.g. Baud rate, interface selection OFF OFF Parameter 1 (mainly driver parameter AC) ON ON Parameter 2 (mainly driver parameter AD) OFF ON Parameter 3 (mainly driver parameter AF) ON ON Parameter 4 (mainly driver parameter AF) For details refer to driver manual PCS 91.xxx
DIL 7	=	Configurations baud rate PCS 009, PCS 090, PCS 095, PCS 095.1 ON = 115.0 kBaud OFF = 38.5 kBaud
DIL 8	=	Operation Mode ON = stop, service programme expected OFF = rund, normal operation Warning !! This switch must be set to OFF during operation, otherwise malfunctions may occur in the PCS and in the programmable controller !!
DIL 9	=	write protection EEPROM ON = EEPROM re-writable OFF = EEPROM write protected
DIL 10	=	Contrast display adjustable with the HLP- and +/- key ON = alteration possible OFF = not possible

The DIL switch 9 should be switched to off after OFF after programming, otherwise the data content can not be guaranteed under all circumstances. In normal circumstances (including on/off switching at any time) there is no chance whatever of data loss.

The contrast normally only has to be adjusted once, it should be put in the OFF position after the setting of the DIL switch 10.

2.2.2 LED DISPLAYS

Every light display can be in 4 different states: OFF, ON, FLASHING and RAPID FLASHING. The FLASHING state is made up of 75% bright phase and 25% dark phase, the condition rapid FLASHING consists of 75% dark phase and 25% bright phase.

The green and yellow LEDs at the function keys are available for the PLC to change. They are controlled by the LED status W10 und W11. The LEDs additionally available via the function keys F9...F18 for the PCS 095, PCS 095.1 are controlled by the extra LED status, words W24 and W25.

The 2 green and 1 yellow LEDs to the right of the control keys show the state of running of the PCS.

INPUT REQUIRED	MENU, INFORMATION WARNING, FAULT	COMMUNICATION FAULT
?	!	ERR
green	green	yellow

■ **(?) INPUT REQUIRED**

ILLUMINATED The PCS is waiting for key activation (quit, i.e., delete from message, input of nominal values, closing of a menu)

FLASHING If a message with cancel mode 4 is shown in the display, this LED flashes as long as the corresponding message bit is log 1 (the message can not be deleted). If the message bit is 0, then it is continually illuminated and the message can be cancelled with CLR. Should the HLP key be pressed and a help text is programmed to the currently activated priorities, this LED flashes alternately with the (!)-LED.

■ **(!) MENU, INFORMATION, WARNING, FAULT**

ILLUMINATED An INFORMATION, a WARNING and a FAULT are shown in the display.

FLASHING A MENU, a WARNING, an INFORMATION or a FAULT is switched on, however is not shown owing to an activated order of priorities in the command word A (W13; Bit 8...11) (at the moment). Should a help key be pressed down and a help text is programmed to the currently activated priorites, this LED (!) flashes alternately with the (?)-LED.

■ **(ERR) COMMUNICATION FAULT**

ILLUMINATED The communication has not been started since the switch on.

FLASHING The communication to the PLC has been broken.

When normal communication is taking place this LED of OFF. Should the communication be interrupt (after it had just been functioning) the acoustic alarm is activated for a short time and the LED begins to flash.

Warning!
Check the action/reaction of the programmable controller!
The action/reaction of the programmable controller has to be checked after a restart of the programmable controller following a communications loss.

■ **CURSOR KEY LEDs IN MENUS**

In this mode the (!)-LED is off or flashing. The arrow keys LED are enabled via bit 5 in command word A (W13). ARROW KEYS-LED

ILLUMINATED: Further nominal values which can be edited can be reached with this arrow.

FLASHING: Activation of this arrow key enables this menu node to be left.

■ **CURSOR KEYS LEDs IN MESSAGES**

The (!)-LED is on, the arrow keys LEDs are enabled via the bit 14 in command word A (W13). Illuminated

ARROW-UPWARDS:	The main lines of this message can be activated.
ARROW-BELOW:	The follow-on pages of this message can be displayed.
ARROW-LEFT:	The manual scrolling is enabled and can be switched over to previous messages.
ARROW-RIGHT:	The manual scrolling is enabled and can be switched over to later messages.

■ **CURSOR KEYS LEDs IN HELP TEXTS**

In this mode the (!)-LED flashes alternately with the (?)-LED. The arrow keys LEDs are enabled via bit 15 in the command word A (W13). Illuminated

ARROW-UPWARDS:	The main lines of this help text can be activated.
ARROW-BELOW:	The follow-on pages of this help text can be displayed.

2.2.3 DISPLAY AND CONTRAST ADJUSTMENT

When the PCS is in operation there are backlit lines (PCS 009: 4 lines x 20 characters, PCS 090: 2 lines x 40 characters, and PCS 095, PCS 095.1: 4 lines x 40 characters) . The character set is limited to the latin character set, including a few special characters. National special characters (eg. ä, ö, ü, ß) must be created via the character programme. For this purpose there are 8 characters to choose from. A character table can be found in the forward of this manual.

Flashing of individual characters (nominal value input) is administered by the PCS itself. Operating texts can flashed when used as default text through bit 15 log 1 in command word B (W14). With message texts this can follow for every priority via bit 8...10 in the command word A (W13). This switch over is also possible form the PLC at any time.

The contrast of the display characters can be altered on mass. The key HLP together with the key + increases the contrast of the characters, the keys HLP and - reduce the contrast until the script has almost completely disappeared. The setting is retentive, i.e., the very last ajustment remains stored even after switching off the PCS. To avoid an error of adjustment to the contrast, the adjustment can be disabled with the DIL switch = OFF.

2.2.4 KEYS

They are divided into function keys, numerical keypad and control keys. All keys are made available as made available as key bits in the PLC. As long as a key is activated, a log 1 appears in the corresponding bit of the word range. The "pressing" of a key sets off a short acoustic signal, the so called keyboard click. Some keys also reproduce repeating acoustic signals on account of their "REPEAT" function.

The function keys F1 to F8 for the PCS 009 and PCS 090 and F1 to F16 for the PCS 095 and PCS 095.1 are only transmitted to the programmable controller. They have no internal functions.

The numerical keypad and the control keys also have PCS internal functions each depending on the displayed priority and are therefore to be interpreted in the PLC with caution.

Priority 0 = DEFAULT TEXT: In this instance the HLP key has internal functions.

Priority 2 = MENU: In this case the numeric keypad 0...9 as well as the control keys +,-,.,Arrow, CLR, ENTER and HLP, internal functions.

Priorities 4 to 8 = MESSAGE PRIORITIES: depending on the programming of the PCS (cancel mode number of message text lines, message help texts), the ARROW keys and the CLR and HLP key each have an internal function.

Priority 12 = HELP: On this occasion HLP, as well as ARROW-UPWARDS and ARROW-DOWN each have internal functions when more than one display is registered.

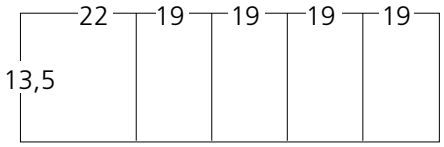
On activating non-permissible keys, exept for priority 0 = default text (only HLP key), the acoustic fault message rings out. Should the priority be limited by blocking the priorities 4...8 in the command word A (W13) to priority 0 = default, the numerical keypad as well as the control keys (exception: HLP) can be occupied with special machine functions. It is to be observed that the priority 12 = HELP is not lockable.

If the acoustic fault message should prove annoying, it can be switched off with bit 4 in the command word A (W13) = logic 1. At the same time the "REPEAT" click will be suppressed.

2.2.5 INSCRIPTION FIELD

An individually design foil for labeling the F-keys can be inserted into the inscription field. For the PCS 009 the foil to be inserted should have the following dimensions:

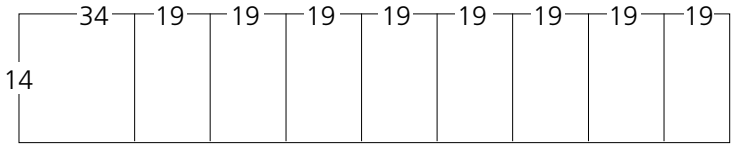
Length: 98 +0 - 0.4 mm (left margin = 22 mm)
Width: 13.5 +0 -0.4 mm



Thickness of the cover foil: max. 0.1 mm. 0.9 mm are covered at the top and bottom margin. The visible window for each function key measures 15 mm (horizontal) x 12 mm (vertical).

An individually design foil for labeling the F-keys can be inserted into the inscription field. For the PCS 090 the foil to be inserted should have the following dimensions:

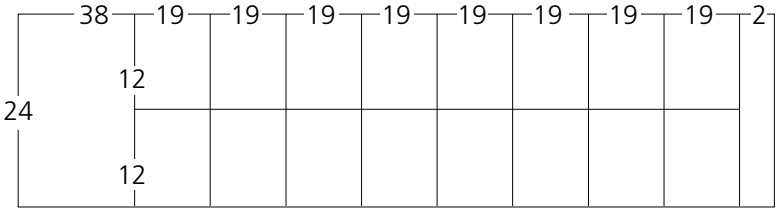
Length: 186 +0 - 0.4 mm (left margin = 34 mm)
Width: 14 +0 -0.4 mm



Thickness of the cover foil: max. 0.1 mm. 0.9 mm are covered at the top and bottom margin. The visible window for each function key measures 15 mm (horizontal) x 12 mm (vertical).

For the PCS 095, PCS 095.1 the foil to be inserted should have the following dimensions:

Length: 192 +0 - 0.4 mm (left margin = 38 mm, right margin = 2 mm)
Width: 24 +0 -0.4 mm



Thickness of the cover foil: max. 0.1 mm. 1.75 mm are covered at the top and bottom margin. The visible window for each function key measures 15 mm (horizontal) x 11.6 mm (vertical).

2.2.6 ACOUSTIC SIGNAL

3 acoustic signals are available.

- a short keyboard click on pressing a key.
- when a key with a "REPEAT" function is "pressed", a "REPEAT" sound is heard.
- a 0.5 second duration acoustic fault message after having pressed a false key.

The volume of the acoustic signal can be adjusted on the rear side of the PCS by means of a potentiometer.

Should the acoustic ringing of a fault message be annoying, then it can be turned off with the word 13 bit 4 = logic 1. At the same time the "REPEAT" sound is suppressed.

2.3 CONNECTIONS

2.3.1 OPERATING VOLTAGE

The connections for the operating voltage are fixed as screw terminals for wires up to 2 mm^2 diameter. For more about power consumption and limits of operating voltage read the chapter "Technical Data".

Warning!
The protective conductor and 0V of the supply voltage are separated in the device. The protective conductor is also connected to pin 1 of the serial interfaces (except for the noise filter). The enclosure must be grounded to avoid noise in the best way. Additionally, 0V must be neutralized near the power supply (according to VDE regulations).

2.3.2 SERIAL INTERFACES

The PCS 009, PCS 090, PCS 095, PCS 095.1 feature a combination interface. Only one interface can be used at a time. On the 25 pol D-type there is either an RS 232 (V24) or alternatively a TTY (line current interface), active or passive, available. On the 15 pol D-plug an RS 422 or alternatively an RS 485 interface is available. With regard to this please take note of the driver manuals PCS 91.xxx.

With a PLC coupling through the RS 422/RS 485, the programming cable PCS 733 can be plugged in at the same time. During the configuration of the PCS the interface RS 422/RS 485 is switched to high resistivity.

2.3.2.1 RS 232/TTY INTERFACE

2.3.2.1.1 CONFIGURATION/PROGRAMMING

With the help of the RS 232 interface you can establish the configuration/programming of the PCS 009, PCS 090, PCS 095, PCS 095.1 from a PC/PG (also refer to PCS*PRO*) with the programming cable PCS 733 (configurations cable). The start up to the configuration, i.e., programming is observed at the DSR input. The PCS is thereby ready for programm transfer. Please note that in order to programm, the EEPROM must be enabled with the DIL switch 9 = ON.

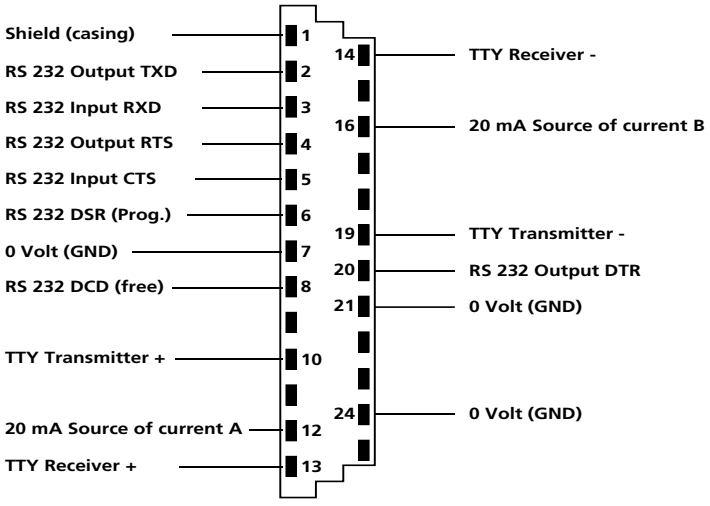
Attention!
The level at DSR (pin 6) is determined by the PC output DTR (25-pole: pin 20; 9-pole: pin 4). Since the level of this pin is not defined after booting the PC/ programmer or after exiting a program, it is possible that the PCS is in configuration mode (only if the programming cable PCS 733 is plugged in). In this case, the PCS program is stopped. Any communication with the programmable controller will be aborted. In this case, you must disconnect the PCS 733 cable. The PCS*PRO* software sets the correct level at this pin.

2.3.2.1.2 COMMUNICATION

Depending on your driver and the PLC being used, you need to utilise a special communication cable. Furthermore DIL switches 5 and 6 must be set according to the programmed driver parameter. For information regarding this please refer to the respective driver manual PCS 91.xxx.

2 seperate line current sources (A+B) are at the disposal of the TTY.

(Overhead view of the plug)



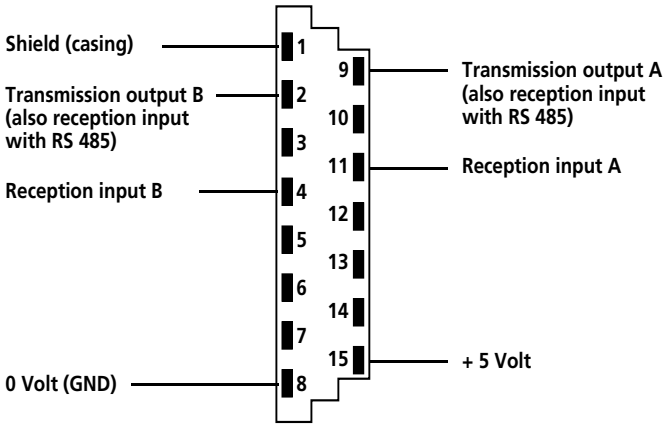
Warning!
If external current loop sources are used, the maximum e.m.f. may not exceed 15 V. Furthermore, real current sources with a maximum of 22 mA are required. Otherwise malfunctions may occur in the PCS and in the programmable controller!

2.3.2.2 RS 422/RS 485 INTERFACE

You will need a special communication cable depending on the driver and the PLC that you use. In addition the DIL switches 5 and 6 must be set according to the programmed driver parameters PCS 91.xxx.

This interface is intended for the communication only.

The RS 422 communication utitises the pins 2 and 9 for transmission and pins 4 and 11 for reception. On the other hand RS 485 applies pins 2 and 9 to transmit and receive. For further details refer to the "PCS 91.xxx.Driver Manual".



Warning!
Check the action/reaction of the programmable controller!
The action/reaction of the programmable controller has to be checked after a restart of the programmable controller following a communications loss.

2.4 OPERATING THE PLC

This section contains that information which is required for the entire planning of the operation concept. It is not however considered as a replacement for the specific machine documentation. The end user will require this in addition.

2.4.1 VARIABLES/TEXTS/MENUS

Variables are inserted in a text. A maximum of 4 variables can be defined per line of text. These variables are normally found in the PLC as from word 30. Depending on their type they are presented in the PCS in a written as well as a numerical form. One differenciates here between ACTUAL-, NOMINAL- AND NOMINAL-P values.

ACTUAL values are variables which may not be altered by the PCS. They are purely presented as values in the PCS display.

Nominal and nominal-P values are variables, which can be both displayed by the PCS as well as altered by it. You'll find the altered written value in the word address under the variable definitions which you yourself wrote. A key function can be reached with the addition (-P). For example it is possible to allow only specific users access to specific nominal values. Should the nominal value-P variables be barred (in word 14 bit 7 on logic 0), these variables will then be displayed as actual values and cannot be altered.

The text differenciates itself between operating, information and help texts.

Operating texts are applied in the priority 0 (default text), as well as in priority 2 (menus). They are 2 lines with the PCS 090 and 4 lines with the PCS 009, PCS 095, PCS 095.1.

In respect of the programming, message texts are applied as information (priority 4), warnings (priority 6) or faults (priority 8). They may have a maximum of 32 lines.

As far as they are configurated, help texts appear for the priorities 0, 2, 4, 6 and 8, on pressing the HLP key. As soon as the key is released, this text disappears. A help text is a maximum of 32 lines similar to the message text.

Menus are collections of "nodes". Every node is allocated to an operating text number. One node of a menu is the so called start node. The operating text defined with this node appears first in the display on starting the menu with word 14, bit 0...7. It is possible with these nodes by means of individual definitions via the "Arrow keys", to branch out into other nodes. As long as bit 5 is in command word A (W13) logic 1, there is the possibility of branching out as displayed in the arrow keys LEDs.

Every variable must be defined before being used in a text. In the same way, texts which are to be applied in menus, have to be previously defined.

2.4.1.1 VARIABLES

Variables can be applied to every text. From this position the PCS reserves room for the variables. The display form and the length are not needed in the variable description. Maximum 4 variables can be used per text line (with the application of the ASCII variable, only one variable per line is permissible). When writing text, the additional variable lengths in each line have to be taken into consideration. Use the programming software "PCS*PRO* ", as this automatically takes into consideration the maximum variable lengths when defining the texts.

A difference is made between INTERNAL and EXTERNAL variables. The source values of the EXTERNAL variables lie in the PLC. A variable definition must be written for these variables. The description of the external variables is filed in the configuration of the PCS. With respect to the internal variables, this is already to hand.

In addition the variable types (V)BIN(0)-1,A permit scaling. That means a given range of values (source range) in the PLC will be displayed in another display range (target area) in the PCS (restrictions: the multiplicator must be positive!).

The number of the pre- and after decimal point positions with every BIN (binary), as well as limiting values; that is minimum and maximum value; are programmable as constants.

BCD(0)-1,2 allow the definition of a minimum and a maximum value, as well as a definable mantissa (digits).

Every variable can be defined as an ACTUAL-, NOMINAL- or NOMINAL-P value.

2.4.1.2 EXTERNAL VARIABLE FORMAT

1. BIT:
A character string (inscription) is allocated to the two possible conditions of the bit in the PLC. The character string can be freely selected and is allowed to be the maximum length of a display line, and that is 40 characters. It cannot contain any variable itself. The place reserved determines the length of both inscriptions. The BIT variable is written into the PLC immediately with each alteration.

2. STRING:
A character string (inscription) can be allocated to each value of the lowest value byte of a word in the PLC. There upon the maximum number of inscriptions is 256. The maximum length of an inscription is equal to one display line which is 40 characters. The reserved place is the length of the longest inscription. The character string itself must not have any further variables.

3. CSTRING
A character string (inscription) can be allocated to each value of the lowest valued byte of a word in the PLC. Thereby the maximum number of inscriptions is 256. The maximum length of an inscription is equal to one display line, that is to say 40 characters. The reserved place is the length of the longest inscription. The character string itself must not have any further variables. The CSTRING variable differs from the STRING variable in that it is stored in the PLC after each alteration, without need for enter key.

4. BCD:
These are values displayed whose number of digits can be selected. These digits must be present in the PLC as a BCD format. The inclusion of a decimal point is not possible. Digits that are not used are ignored on reading of the ACTUAL value and are reset on writing the nominal value (-P-). The following variable formats are possible.

BCD-1: Possible number of digits between 1 and 4. This variable requires a word in the PLC.

BCD0-1: As in BCD-1, however pre-zeroes are displayed instead of empty spaces.

BCD-2: Possible number of digits between 1 and 8. This variable requires a double word in the PLC.

BCD0-2: As BCD-2, however pre-zeroes are displayed instead of empty spaces.

5. BIN:
The 16 bit value of a word or 32 bit value of a double word in the PLC are displayed in the fixed point format as a non pre-character figure. The variable requires maximum 11 digit places (with decimal points). The inclusion of the decimal point is made possible with the selection of the pre- and after point position. At the same time the position for the decimal point in the display is also to be condisdered. With 16 bit variables it is also possible to have scaling, that is a conversion of the range of values PLC -> PCS and in the reverse PCS -> PLC. The display range in the PLC with the 16 bit variables is between $0 and $FFFF, and with 32 bit variables between $0 and $FFFFFFFF. The range of values presentable in the PCS is between 0 and maximum 4 294 967 295. The following variable formats are possible:

BIN-1: This variable occupies a word in the PLC. The number of pre-decimal point is definable between 1 and maximum 10. The number of the after decimal point positions is between 0 (without decimal point) and maximum 9. As soon as the after decimal point positions are declared, the variable requires one further character position in order to superimpose the decimal point. If the minimum value of the PLC is different from the minimum value of the in the PCS, i.e., the maximum value of the PLC from that of the PCS, then it is dealing with a scaling BIN variable. With this type of variable, the input of the pre-decimal point position is seperate from the after decimal point positions, should an after decimal point position have been given. Activating the (.) key puts in the after decimal point positions. This kind of figure input is also known as pocket calculator input.

BIN0-1: As in BIN-1, though here pre-zeroes instead of empty spaces are displayed.

BIN-A: As in BIN-1, however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left).

BIN0-A: As in BIN-1, however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left). Also here pre-zeroes instead of empty spaces are displayed.

BIN-2: This variable occupies a double word in the PLC. The number of pre-decimal point is definable between 1 and maximum 10. The number of the after decimal point positions is between 0 (without decimal point) and maximum 9. As soon as the after decimal point positions are stated, the variable requires one further character position in order to superimpose the decimal point. With this type of variable, the input of the pre-decimal point position is separate from the after decimal point positions, should an after decimal point position have been given. Activating the (.) key puts in the after decimal point positions. This kind of figure input is also known as pocket calculator input.

BIN0-2: As in BIN-2, though here pre-zeroes instead of empty spaces are displayed.

BIN-B: As in BIN-2, however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left).

BIN0-B: As in BIN-2 however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left). Also here pre-zeroes instead of empty spaces are displayed.

BIN0-B: As in BIN-2 however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left). Also here pre-zeroes instead of empty spaces are displayed.

6. VBIN:

The 16 bit value of a word or 32 bit value of a double word in the PLC are displayed in the fixed point format as a non pre-character figure. The variable requires maximum 12 digit places (essentially with operational sign and alternatively with decimal points). The superimposition of the decimal point is made possible with the selection of the pre- and after decimal point position. At the same time the position for the decimal point and the operational sign in the display is also to be considered. With 16 bit variables it is also possible to have scaling, that is a conversion of the range of values PLC -> PCS and in the reverse PCS -> PLC. The display range in the PLC with the 16 bit variables is between $8000 and $7FFF, and with 32 bit variables between $80000000 and $7FFFFFFF. The range of values presentable in the PCS is between -2 147 483 648 and maximum +2 147 483 647. The operational sign can be altered with the help of "+" or "-" key. The following variable formats are possible:

VBIN-1: This variable occupies a word in the PLC. The number of pre-decimal point is definable between 1 and maximum 10. The number of the after decimal point positions is between 0 (without decimal point) and maximum 9. As soon as the after decimal point positions are stated, the variable requires one further character position in order to superimpose the decimal point. If the minimum value of the PLC is different from the minimum value of the in the PCS, i.e., the maximum value of the PLC from that of the PCS, then it is dealing with a scaling VBIN variable. With this type of variable, the input of the pre-decimal point position is separate from the after decimal point positions, should an after decimal point position have been given. Activating the (.)-key puts in the after decimal point positions. This kind of figure input is also known as pocket calculator input.

VBIN0-1: As in VBIN-1, though here pre-zeroes instead of empty spaces are displayed.

VBIN-A: As in VBIN-1, however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left).

VBIN0-A: As in VBIN-1, however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left). Also here pre-zeroes instead of empty spaces are displayed.

VBIN-2: This variable occupies a double word in the PLC. The number of pre-decimal point is definable between 1 and maximum 10. The number of the after decimal point positions is between 0 (without decimal point) and maximum 9. As soon as the after decimal point positions are stated, the variable requires one further character position in order to superimpose the decimal point. With this type of variable, the input of the pre-decimal point position is separate from the after decimal point positions, should an after decimal point position have been given. Activating the (.) key puts in the after decimal point positions. This kind of figure input is also known as pocket calculator input.

VBIN0-2: As in BIN-2, though here pre-zeroes instead of empty spaces are displayed.

VBIN-B: As in VBIN-2, however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left).

VBIN0-B: As in VBIN-2, however the value is not entered according to the pocket calculator input, but by means of an interjection over the decimal point (from right to left). Also here pre-zeroes instead of empty spaces are displayed.

7. WORD

The 16 bit value of a word in the PLC is displayed in the bit format. The keys + and - enable a cursor to be positioned on the individual bits. An individual bit be cancelled with the 0 key or set with 1 key. This data format requires a definite 17 character place in a line. An empty space has been set between the HIGH- and the LOW-byte as an convenient division.

The WORD variable of the PCS is used to represent the content of a 16-bit word in different formats:

KM - bit-by-bit representation of a word, e.g. '10001001 10101011' (see word variable)
KH - hexadecimal representation of a word, e.g. '89AB' (for entry see ASCII variable)
KY - byte-by-byte decimal representation, e.g. '137 171' (for entry see binary variable)

Attention!
The variable formats KM, KH, and KY are available starting with the following hardware versions:
PCS 009 version V2000 and up
PCS 090 version V205B and up
PCS 095/095.1 version V4067, 4 data records
PCS 095/095.1 version V5066, 1 data record

8. ASCII

Up to 16 characters (8 words) in the PLC can be displayed or altered as ASCII characters. The + and - keys enable the ASCII characters to be presented with the next higher or lower ASCII code. The (.) switches the cursor one position to the right. After the last character has been entered, activation of the (.) key, the cursor again appears on the 1st character.

9. TIMER

The TIMER variable format is used to specify a 3-digit time value and to select the time base from 4 possible values.

The TIMER variable reads/writes the content from/into a 16-bit word in the following format:

'00dd cccc bbbb aaaa'

aaaa = BCD coded number D1 (0..9) of the time value
bbbb = BCD coded number D2 (0..9) of the time value
cccc = BCD coded number D3 (0..9) of the time value
dd = Time base value (0..3)

Content of the word '2 1 0 0' - time value 100 corresponds to 100 seconds
|
Time base 2 (corresponds to * 1s)

The texts used to represent the selected time base can be setup as desired. To modify a TIMER preset value, the time value and (if required) the time base must be modified. To switch between these two entries, use the (.) key of the PCS. The time value can be directly modified using the numeric keys. If the time base modification is activated, it can be selected with the (±) key.

Attention!
The variable format Timer is available starting with the following hardware versions:
PCS 009 version V2000 and up
PCS 090 version V205B and up
PCS 095/095.1 version V4067, 4 data records
PCS 095/095.1 version V5066, 1 data record

2.4.1.2.1 VARIABLES FORMAT BIT

It is assumed that you have created a bit variable on word 30 as nominal value with the assistance of the programming software PCS*PRO* . You have selected bit 15 as bit number. You have programmed the character string (inscriptions) for the logic bit condition 0 with "CLOSED" and for the logic bit condition 1 with "OPEN".

Summary:

Word number:	30
Class:	NOMINAL
Variable format:	bit
Bit position:	15
Inscription 0 (AP0):	CLOSED
Inscription 1 (AP1):	OPEN

The variable is incorporated into the operating text 0 as follows:

VALVE 0 IS IN ▯▯▯▯▯▯▯▯▯▯▯▯ CONDITION

If the bit 30.15 = 0, there appears with the selected operating text 0 in the display:

VALVE 0 IS IN THE CLOSED CONDITION

If the bit is 30.15 = 1, there appears with the selected operating text 0 in the display:

VALVE 0 IS IN THE OPEN CONDITION

If this operating text 0 is utilised in the menu node, then the bit 30.15 can be set with the key + and set back with the key -. The alteration is carried out immediately after every activation of the keys. The remaining bits of the word 30 are not influenced by writing back.

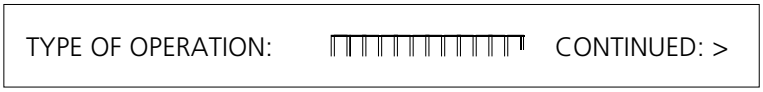
2.4.1.2.2 VARIABLES FORMAT STRING

It is assumed that you defined a STRING variable on word 31 as nominal-P value with the help of the programming software PCS*PRO* . The character strings (inscriptions) 0...2 are programmed with "SERVICE", "SETTING UP OPERATION" and "AUTOMATIC OPERATION".

Summary:

Word number:	31
Class:	NOMINAL-P
Variable format:	STRING
Inscription 0 (AP0):	SERVICE
Inscription 1 (AP1):	SETTING UP
Inscription 2 (AB2):	AUTOMATIC

The variable is inserted in the operating text 15 as follows:



If the value 1 is present in the lower value byte of word 31, then with active operating text 15 there appears in the display.

If the variable is used in a menu, the value in the word 31 can be decremented with keys "-" until the value 0 and incremented with the key "+" to the value 2. However bear in mind that an altered value is written back into the word first after "ENT" or departure from the variable field. If the value is to be written at once into the PLC, refer to CSTRING.

ATTENTION:

1. The bits in the higher valued byte of word 31 are ignored on reading; on writing them back into the PLC, they are set to 0. This is a means of assistance to establish alterations brought about by the PLC programme.
2. Should the old value not be altered, then it won't be written back (even including Bits 8...15).
3. A maximum of 256 inscriptions are allowed (including 0).
4. The limitations set themselves according to the number of programmed inscriptions; whose minimum value is 0.
5. At least 3 inscriptions must be defined, otherwise the variable is to be declared as a bit.
6. It is impossible to leave the input field with a value outside the limiting values as soon as editing has begun.
7. A restoration of the original value is possible at any time with the "CLR" key.

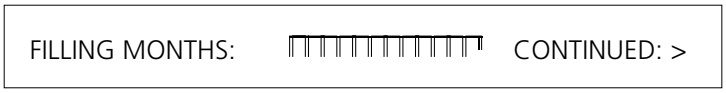
2.4.1.2.3 VARIABLES FORMAT CSTRING

It is assumed that you defined a CSTRING variable on word 32 as nominal-P with the help of the programming software PCS*PRO* . The character strings (inscriptions) 0...11 are programmed with "JANUARY", "FEBRUARY", "MARCH", "APRIL", "MAY" until "DECEMBER".

Summary:

Word number:	32
Class:	NOMINAL
Variable format:	CSTRING
Inscription 0 (AP0):	JANUARY
Inscription 1 (AP1):	FEBRUARY
up to Inscription 11 (AB11):	DECEMBER

The variable is inserted in the operating text 20 as follows:



If the value 5 is present in the lower value byte of word 32, then with active operating text 20 there appears in the display.

If the variable is used in a menu, the value in the word 32 can be decremented with key "-" until the value 0 and incremented with the key "+" until 11 (=$000B). A modified value is written into the PLC at once. This is contrary to that with CSTRING.

ATTENTION:

1. The bits in the higher valued byte of word 32 are ignored on reading; on writing them back into the PLC, they are set to 0. This is a means of assistance to establish alterations brought about by the PLC programme.
2. Should the old value not be altered, then it won't be written back (even including Bits 8...15).
3. A maximum of 256 inscriptions are allowed (including 0).
4. The limitations set themselves according to the number of programmed inscriptions; whose minimum value is 0.
5. At least 3 inscriptions must be defined, otherwise the variable is to be declared as a bit.
6. It is impossible to leave the input field with a value outside the limiting values as soon as editing has begun.
7. A restoration of the original value is **not** possible with the "CLR" key.

2.4.1.2.4 VARIABLES FORMAT BCD

The BCD variable formats are divided into the following sub groups:

Variable type	16 bit	32 bit	Number of digits	Pre-zeros
1. BCD-1	x		1...4	
2. BCD0-1	x		1...4	x
3. BCD-2		x	1...8	
4. BCD0-2		x	1...8	x

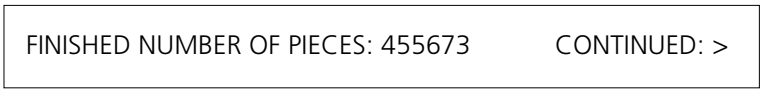
It is assumed that you have defined a BCD variable (BCD-2) on word 33 as nominal-P value with the assistance of the programming software PCS*PRO* . You will to display 8 digit positions, the typed in minimum value should be 90 and the maximum value 50000000.

Summary:

Word number: 33
Class: NOMINAL-P
Variable format: BCD-2
Inscription 0 (AP0): 8
Inscription 1 (AP1): 90
up to Inscription 11 (AP11): 50000000

The variable is inserted into the operating text 100 as follows:

If the value $0045 (69) is in word 33 and the value $5673 (22131) is in word 34, then there appears in the selected operating text 100 in the display:



The 2 pre-zeros are not shown as this is the variable format BCD-...! If you wish the pre-zeroes to be displayed, put in the variable format BCD0-... instead of BCD-...

ATTENTION:

1. Unnecessary higher value bits will be ignored and written back to 0.
2. Scalling and the superimposing of decimal points is not possible.
3. Intermediate values will not be recognised. The writeback first occurs after "ENT" or on leaving the variable field.
4. Offsets are also possible: "1", "0", "+", would give the intermediate result of 455683 in the above example as a result. As this case is about an intermediate result, no writeback will be made (although the cursor now refrains from blinking)!
5. It is impossible to leave the input field with a value outside the limiting value after editing has begun.
6. You can also increment and decrement with the operational sign keys (with auto repeat).
7. It is possible to bring back the previous values at any time with "CLR".

2.4.1.2.5 VARIABLES FORMAT BIN

The BIN variables format are divided up into the following sub-divisions as follows:

Variable type	16 bit	32 bit	Pocket calculator entry	Scaling	Operational sign	Pre-zeros
1. BIN-1	x		x	x		
2. BIN-A	x			x		
3. BIN-2		x	x			
4. BIN-B		x				
5. VBIN-1	x		x	x	x	
6. VBIN-A	x			x	x	
7. VBIN-2		x	x		x	
8. VBIN-B		x			x	
9. BIN0-1	x		x	x		x
10. BIN0-A	x			x		x
11. BIN0-2		x	x			x
12. BIN0-B		x				x
13. VBIN0-1	x		x	x	x	x
14. VBIN0-A	x			x	x	x
15. VBIN0-2		x	x		x	x
16. VBIN0-B		x			x	x

The difference between (V)BIN(0)-1 i.e., (V)BIN(0)-2 and (V)BIN(0)-A i.e., (V)BIN(0)-B is in the type of editing applied:

- (V)BIN(0)-1, (V)BIN(0)-2: Pocket calculator input with separate pre-decimal point and after decimal point input (only when after decimal points are available). Shift is achieved with the "." key.
- (V)BIN(0)-A, (V)BIN(0)-B: Pushing through from right to left (via decimal point). The "." key has no function.

Variables 9...16 (V)BIN0-... are only definable with the programming environs PCS*PRO*!

It is assumed that with the assistance of the programming software PCS*PRO*, you have defined a BIN variable (BIN-) on word 34 as nominal value. You want to present and key in two pre-decimal point and one after decimal point. In addition you want to incorporate scaling. Values of between 0 and 100 (0 and 10,0) may be typed into the PCS. This range of values should be sent to the PLC however as 0...4095 ($0...$0FFF). Pre-positioned zeros should be suppressed.

Summary:

Word number:	34
Class:	NOMINAL VALUE
Variable format:	BIN-1
Pre-decimal point positions:	2
After decimal point positions:	1
Minimum value PCS:	0
Maximum value PCS:	100
Minimum value PLC:	0
Maximum value PLC:	4095

The variable is to be inserted in the operating text 120 as follows:

POTENTIAL: ⌈⌈⌈⌈ VOLT CONTINUED: >

If the value $0800 (2048) is in word 34, then there appears in the display with operating text 120 the following:

POTENTIAL: 5.0 VOLT CONTINUED: >

Operation as nominal variable in a menu:

- The value can be altered with the numeric keys.
  (V)BIN(0)-1(2): Separation of pre-decimal point and after decimal point, change occurs with key (.).
  (V)BIN(0)-A(B): Simple pushing through from right to left springing over the decimal point.
- Offset input possible (not with VBIN variables!): e.g., ".", "2", "+", : new display (example) 5.2!
- Keys "+"/"-":
  BIN(0)-1,2,A,B: Adding / subtracting is with 1 (also with ".").
  VBIN (0)-1,2,A,B: Change of operational signs at any time.

Attention!
- Only altered values are written back within the limiting values.
- If the original value is outside the limiting values, then inverse fields will be displayed.
- If a value outside the limiting values has been keyed in (this is only possible with direct numeric input) then by ENTER of departure from the field, a check will be carried out. If there is an error and the given value was smaller than the minimum value, then the minimum value will be displayed. Furthermore the acoustic signal rings out and nothing is written in the PLC.
- It is possible to leave the inverse field. For example, if the first variable in a menu text is outside the limiting value, then it is not possible to page further. First of all the value has to be corrected (a valid value is keyed in when BIN is "+", "-" or "CLR" and with VBIN only "CLR" or per character input).
- The declared range of values (PLC and PCS) may only be negative with VBIN(0) variables. In this case the minus sign is merely to be set before the corresponding value or values.

2.4.1.2.6 VARIABLES FORMAT WORD

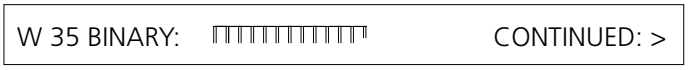
It is to be assumed that you have defined a word variable on word 35 as a nominal value with the help of the programming software PCS*PRO* .

Summary:

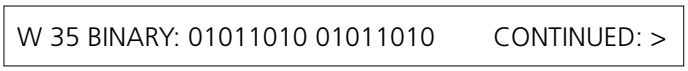
Word number:	35
Class:	NOMINAL VALUE
Variable format:	WORD

1. Format when using bit-by-bit representation (corresponds to KM)

The word on the specified address is represented in binary format using 0 and 1 (e.g. a PRESET value has been assigned to word 135): The insertion of the variable into the operating text 99 is represented below:



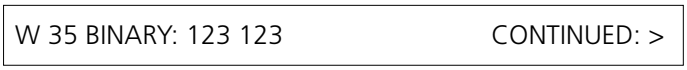
If word 135 contains the value $5A5A the following is displayed with operating text 99 selected:



Using the + and - keys, the cursor can be moved bit-by-bit if the variable is used in a menu. The bit at the cursor position can be set to logic. 0 and 1 by using the 0 and 1 keys.

2. Format when using the dual decimal representation (corresponds to KY)

The word on the specified address is represented using decimal numbers with separation of the high and low byte of the word:



0..9: calculator entry of high/low byte; Point: switching between digit high/low byte; +/-: INC/DEC of high/ low byte

3. Format when using hexadecimal representation (corresponds to KH)

The word on the specified address is represented word-by-word using the numbers 0...F.

Point: change to the next digit (right direction); 0...9: assigning a number to each digit; +/-: accessing the numbers A..F (pseudo tetrad).

Generally, a modified value is only stored in word 135, if the ENT key is pressed or if you exit the variable field.

Attention!

*■ If the previous value is not changed, no data are stored.

■ Restoring the previous value is possible at any time using the CLR key.

■ The WORD variable format permanently requires 17 characters in the display (the 8 most significant bits are separated by a SPACE from the 8 least significant bits)!

* This point is only valid, if the operating page options correspond to the default setting!

Attention!
The variable formats KM, KH, and KY are available starting with the following hardware versions:

PCS 009 version V2000 and up
PCS 090 version V205B and up
PCS 095/095.1 version V4067, 4 data records
PCS 095/095.1 version V5066, 1 data record

2.4.1.2.7 TIMER

The variable format TIMER is used to specify a 3-digit time value and to select the time base from 4 possible values.

The TIMER variable reads/writes the content from/into a 16-bit word in the following format:

'00dd cccc bbbb aaaa'

aaaa = BCD-coded number D1 (0..9) of the time value

bbbb = BCD-coded number D2 (0..9) of the time value

cccc = BCD-coded number D3 (0..9) of the time value

dd = Time base value (0..3)

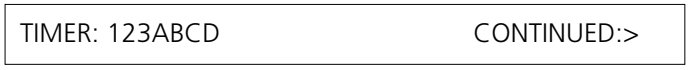
Word content '2 1 0 0' - time value 100 corresponds to 100 seconds
|
Time base 2 (corresponds to * 1s)

The texts used to represent the selected time base can be created as desired. To modify a TIMER preset value, the time value and (if required) the time base must be modified. To switch between these two entries, use the (.) key of the PCS.

The time value can be directly modified using the numeric keys. If the time base modification is activated, it can be selected with the (±) key.

Timer variable in accordance with the Siemens format with 3 BCD digits and 4 Project.AP with a maximum of 37 characters.

Example with an AP comprising 4 characters:



Word format:

Bits 12+13 = These bits indicate the corresponding AP.

Bit 11...0 = 3-digit BCD number

Attention!
The variable formats KM, KH, and KY are available starting with the following hardware versions:

PCS 009 version V2000 and up
PCS 090 version V205B and up
PCS 095/095.1 version V4067, 4 data records
PCS 095/095.1 version V5066, 1 data record

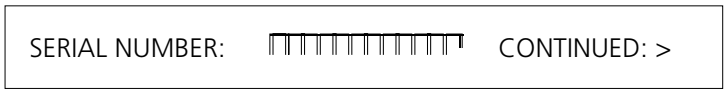
2.4.1.2.8 VARIABLES FORMAT ASCII

It is assumed that you have defined an ASCII variable on word 36 as nominal value with the help of the programming software PCS*PRO* . You wish to be able to key in and display a 16 digit serial number.

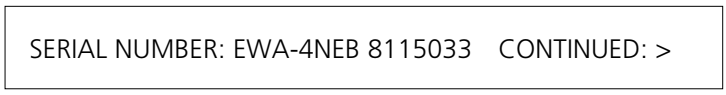
Summary:

Word number:	36
Class:	NOMINAL VALUE
Variable format:	ASCII
Number of characters:	16 (8 words)

The variable is inserted in the operating text 90 as follows:

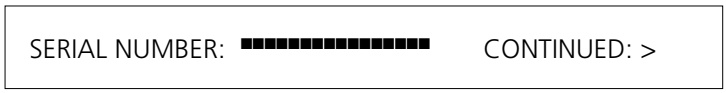


If there exists in the words W36=$4557, W37=$4120, W38=$344E, W39=$4542, W40=$2D38, W41=$3131, W42=$3530 und W43=$3533 (corresponds to the String "EWA-4NEB 8115033), then with the selected operating text 90, the following appears in the display:

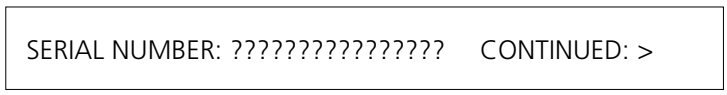


Should the variable be used in a menu, then the cursor (flashing position) can be moved one place to the step by step to the right with the help of the "." key. If the cursor is resting at the variable end (end of the character string), then activation of the "." key sends it once again to the beginning of the variables. Every sign, including the special signs can be selected with the "+" and "-" keys. An altered value is first written in the delivery area first after ENTER or departure from the variables field as from word 36 (W36...W44) Hex coded (except in the case where the value has not been altered).

If the words W36 to W43 are outside the displayable characters, that is to say in the areas $00...$07, $09...$1F or >$7F, there appears in the string display:



After activation of the control keys "+", "-", or "CLR", the "*" characters replaced with "?", so that the variables prediction now consists of 16 characters with $3F "?".



Now the variable can be edited. Afterwards the newly edited variable value can be written in the PLC with ENTER or departure from the variables field.

ATTENTION:
- If the previous value is not modified, no data are stored.
- Restoring the previous value is always possible by pressing [CLR].
- Only 1 ASCII variable may be used for each display line and no other variables may be shown on that line.
- Only even character lengths are admissible!

2.4.1.3 INTERNAL VARIABLES FORMAT

In addition to the user defined variables 6, firmly defined, internal variables are available. At the moment only variables from ZP upwards are used. These variables are ready to be displayed in the PCS-display.

DESCRIPTION OF INTERNAL VARIABLES Z ?				
IDENTIFICATION	DESCRIPTION	TYPE	LENGTH	ACTUAL/NOMINAL
ZP	NUMBER ACTIVE INFORMATIONS	BIN	3	ACTUAL
ZQ	NUMBER ACTIVE WARNINGS	BIN	3	ACTUAL
ZR	NUMBER ACTIVE ERRORS	BIN	3	ACTUAL
ZT	MENU NUMBER	BIN	3	ACTUAL
ZV	SCROLL TIME	BIN	2	NOMINAL
ZX	INTERFACE ERROR	BIN	2	ACTUAL

Here is a short description of the above mentioned variables:

ZP: The number of currently switched on informations are displayed as a 3 digit actual value.

ZQ: The number of currently switched on warnings are displayed as a 3 digit actual value.

ZR: The number of currently switched on errors are displayed as a 3 digit actual value.

ZT: Here the currently active menu number is displayed as a 3 digit actual value.

ZV: Here the scroll time in the message storage can be displayed or altered in seconds. This alteration is only valid until the next RESET and is not guaranteed to be retentive.

ZX: Here the maximum number of faulty (repeat) packets since RESET, are presented. It consists of respectively 100 packets and is a standard for the safety of data transfer. This again depends on the cable length, the type of cable and the amount of electrical and magnet fields of interference. An error rate of up to 1% is safe. This information applies to all the drivers, which support the internal variable ZX.

2.4.1.4 TREATMENT OF VARIABLES

Every variable is automatically read by the PCS; that is to say the specified word number. The PLC specific word number (DW, MW, DM, Counter...) or indication, can be found in the help section of the driver manual PCS 91.xxx. This also applies for nominal values. Here the read value is displayed as preset value (refer to the chapter "Variables in menus").

The following rules apply for the refressing of variables (ACTUAL values or non active nominal-P-values):

- Continual refressing of variables occurs in every priority class. The rate of refress depends on various factors: the number of variables in the display, the type of driver, the transmission speed (baud rate), the number of tasks that can be achieved in a transfer paket, as well as the answer time of the PLC which is independant of the PLC cycle time. The best case shows a refress time of roughly 8 per second.
- There is no difference between internal and external variables. As long as the variable values have not been transfered, spaces are shown in the display. If the read value is outside the filed limiting values in the PCS, then the inverse fields will be displayed in the variables field (every one with dots).
- Nominal-P-variables are treated exactly like ACTUAL values as long as the bit 7 in word 14 is logic 1.

Special treatment is required for PCS 009, PCS 095, PCS 095.1 as follows:

- If there are more than 8 double words from an PLC to read in a display page (4 lines), then the reading is divided into two different PLC cycles. First the variables which are in the first line pair are read, then finally those in the second pair of lines.

The following rules apply for the editing and writing of variables (NOMINAL and NOMINAL-P):

- Nominal-P-variables are first read before activation and then frozen. As a result an alteration of the value by the PLC is not recognisable after freezing. As soon as a key is pressed to editor the nominal values, a flashing cursor appears and the remaining variable is presented in a static form. This doesn't apply with offset input nor with the variable bit or CSTRING, as these are written at once.
- As far as nominal value (NOMINAL or NOMINAL-P) is to be altered, it will be written by activating the ENTER key or by leaving the variables field (permitted arrow key). There is an exception for the menu end. In this case the last presented value is written on any account.
- If an active nominal-P-value is in the display an is set in word 14 bit 7 to zero, then this variable can be written at once. Finally the first to be edited nominal value of this display side will be looked for and presented flashing (not yet edited).
- After a nominal-P-value has been written by the PCS, it will be read twice again (differing PLC cycles). Finally it will be compared with the previous edited value. If there is a difference in the values, the acoustic warning signal rings and the current value of the PLC is momentarily displayed flashing. Thereby a dynamic examination of the limiting values by the PLC is possible. First after activating the proposed value as suggested by the PLC, with ENTER or a permissible "arrow key" can you quit the variable field (or even a menu). With scaled binary variables where the PLC area is larger that the PCS area, care must be taken that the correct value, "level" is presented by the PLC. Here is an example: the range of values of the PCS goes from 0...1000, the range of values of the PLC from 0...65535. The value 10 in the PCS display corresponds to the value 655 in the PLC. The value 11 in the PCS display corresponds to the value 721 in the PLC. If the PLC is written with the value 670, the menu could never be completed as the PCS value of (655) always differs form the 670.

2.4.1.5 TEXT GROUPS

There are 3 groups of freely defined texts:

1. **128 OPERATING TEXTS:** 2 (PCS 090) i.e., 4 (PCS 009, PCS 095, PCS 095.1) line texts, which can be used as DEFAULT TEXTS and MENU TEXTS.
2. **128 MESSAGE TEXTS:** Text pages which can be up to 32 lines in length. These texts are allocated to the message bits and can be displayed as INFORMATION, WARNINGS and as FAULTS.
3. **5 HELP TEXTS:** The HELP TEXT is maximum a 32 line text page, which can be brought anytime into the ON-LINE operation with the HLP key. Individual text pages can be produced according to priority class (default texts, menus, information, warnings and faults).

Additional lines can be found with the "DOWN ARROW" in those texts which have more than 2 (PCS 090) i.e., 4 (PCS 095, PCS 095.1) lines can be switched further. With the "UP ARROW" the first display; known as main lines can be found. If a text consists of only one line, the following lines in the display are empty. If the bit 14 and 15 in word 13 are logic 0, the relevant arrow key LED lights up in order to show whether the main lines or extra lines can be activated.

Alterable texts can be achieved using variables within the main texts. The conversion of numerical and logical values into text form is done by the PCS. The PCS therefore requires a variable definition during programming and also space allocation in the text. This definition contains also the format and the length of the variable. These lengths are important in the formulation of the texts. If the texts are defined with the help of the programming software PCS*PRO* , then the text length is checked automatically.

Apart from monitoring the variables value in the PLC, no extra PLC programme is necessary. The variables are refreshened cyclically in every priority, whereby every value in the display comes from a fixed data exchange cycle. An exception is for the PCS 009, PCS 095, PCS 095.1, where more than eight double words (i.e., 16 words) are read. This reads those variables out of the upper two and lower two display lines in separate PLC cycles.

As the variable can also be presented in text form, the recognition of the variables format BIT, STRING, CSTRING, ASCII, WORD and TIMER important for projecting distribution of text.

Every menu is a collection of between 1 to 255 menu points (nodes). The start up and termination of a menu is controlled by the PLC. Switching between nodes is under operator control only.

2.4.1.6 MENUS/MENU ORGANIZATION

There are a total of 127 menus available. The menus are numbered from 1 to 127. A menu consists of one or more nodes (1...255), an operating text (0...127) is allocated to every node.

The actual node number is displayed in word 6, bit 8...15.

Further nodes can be reached within the menus via the arrow keys. Here the structure is freely programmable. The first specified node is the initial node, i.e., start node. This initial node is displayed by calling the menu through the appointed operating text.

By means of the status of bit 7 from the command word B (W14), the operator can determine at any time whether the nominal-P-variables are alterable or not. If bit 7 = 0, then only a pure nominal value can be altered. If the bit D7 = 1, then the nominal and nominal-P-variables can be altered.

If the cursor is positioned on a nominal-P-value and at the same time bit 7 in word 14 is logic 0, then this nominal value can be changed. When a nominal value is entered withing the appropriate limiting values and transfered to the PLC and read again by the PCS, the editor field is then free. The editing position is initially set on the first nominal value of the display page that is eligibe to be edited. If there are no nominal values, then all variables will be treated as actual values.

2.4.1.6.1 STARTING UP THE MENUS

The PLC programme writes a menu number (1 to 127) on the lower valued byte of the command word B (W14), bit 0...6.

Bit 7 of the command word B (W14) determines whether a nominal-P-value can be altered or not. If bit 7 is logic 0, then this will still be written and finally depart from the variables position, if it is the case that the currently edited nominal value concerns a nominal-P-value.

2.4.1.6.2 TERMINATION OF THE MENUS

Termination of the menus takes place with the PLC, the menu number of the command word B (W14) bits 0...6 are set to logic 0.

The menu can however only bed exited when an altered nominal value has been read twice out of the data area of the PLC and that it is checked with the previously written value. The PLC therefore can recognize and reject (dynamic limiting value examination) locking or minimum and maximum overlapping. Should the nominal value not be taken over by the PLC and therefore immediately written over, then the input field remains active (flashing) with the current PLC variable values. The menu can only be terminated when the comparison of the written nominal value with the read nominal value are checked with one another. In order to show the operator that this nominal value input is not possible an INFORMATION text could be displayed for example. It has to be acknowledged with the CLR key. This acknowledgement doesn't influence the nominal value in any manner (it functions similar to an interruption).

There is an exception for variables, which are presented with inverse fields. In this case, the menu can still be left as long as no editing follows.

The actual menu end can be recognized with the negative edge of bit 0 in word 6 (PCS-status).

2.4.1.6.3 BUILD-UP OF THE MENUS

Each one of the maximum 127 possible menus (1…127) can possess a particular structure. If complex structures are to be used, then it is recommended to procede in the following manner (separate for each menu):

- First the structure is put down on paper, where the node connections are joined with several coloured lines (a different colour for every arrow key).
- Finally an operating text number is allocated to every node. Similar operating texts can be readily used in several menus (furthermore this saves storage space!).
- In conclusion every node is given a particular number (1...255).
- Branches are decided for every node, where every parameter is taken from the sketches. The initial node, also called start up node, must appear first (only when applying software PCS*PRO* . The sequence of the remaining nodes is random (provided they belong to a menu).

Programming of the menu nodes is written in the manual PCS*PRO* . The compiler programme checks the plausibility of the menu definition during the translation. If the programming software PCS*PRO* is used, the syntax control will be already carried out during the editing of the menus. Care is to be taken that menus do not fail. In detail this means that every menu node must be accessible via a path from the start node. There are no further restrictions, i.e., within every node, a given target node within the same menus can be allocated to every arrow key.

When formulating the operating text, it is important to think out a satisfactory operating procedure. It is certainly feasible to consider node points without variables, which only serve the operating procedure. Clarity should be achieved with the application of (programmable) special characters ARROW UPWARDS, ARROW BELOW, as well as ARROW TO THE RIGHT and ARROW TO THE LEFT (refer to character set) (e.g. character $0E = Arrow upwards, $0F = Arrow downwards, <, >).

As long as the arrow key LEDs in the menus (for the PCS 090, PCS 095, PCS 095.1) are free; that is bit 5 of command word A (W13) is logic 0, the operator is additionally guided with optical displays through the menu. If an arrow key lights up statically, it means that another additional variable can be selected within the same menu node. If an LED flashes, activation of this key will cause departure from the currently displayed menu node.

2.4.1.6.4 VARIABLES IN THE MENU

On calling one of the new menus or menu nodes, the first nominal value; after it has been read out of the PLC; will first appear flashing. Should this predetermined value be outside the defined limiting values, then an inverse field instead of a nominal value be displayed. A single activation of the editor key CLR (also + or -) gives a permissible value. This is the maximum or minimum value depending on high or low PLC value. With the ASCII variables "?" is presented as a default value.

Should the sign be altered except for offset input, then the input position is marked by a flashing cursor.

For refreshing variables, the following rules apply: The flashing variable will be picked up once. Every other one on the same display page with nominal and actual values will be continually refreshed. Should the variable be left after an alteration with a flashing cursor, then as early as possible in an PLC cycle and later, the value be read again and compared with the edited value. The input field can be left when the written nominal value is in agreement with the later re-read nominal value. In this way nominal values; dependant on the situation; can be applied within a menu node.

Attention!
As long as the cursor is flashing, the presentation of an intermediate result is taking place. That means that the value in the display is not in agreement with the value in controlling operation!

Numerical values can also be altered in the addition and subtraction mode (also known as offset inputs): <numeric character>, <numeric character>, ...<plus>, possible with BCD(0) and BIN(0)-1,2,A,B. Afterwards the editor is again in the basic condition (variable flashing). It deals likewise here with an intermediate result which cannot yet be written back!

The following rules apply in writing nominal-P-values.

- Basically only altered values are written back (even after ENTER!). If a value is not written, then an acoustic alarmsignal rings. An exception applies in terminating a menu: here the last activated nominal value will at least be written.
- BIT- and CSTRING variables will be written into the transfer area of the PLC with every alteration.
- Should the variables refer to smaller sizes than in the word (as with BCD(0)-1: 1...max. 3 digit, BCD(0)-2: 1...max. 7 digit, STRING and CSTRING), the leading bits will be processed according to the following logic: leading bits are ignored when reading in the predetermined values (i.e., if they are set, then they do not lead to the presentation of inverse fields). On writing back they are set back to zero. This can be evaluated in the PLC for example, in order to be able to react to nominal value inputs.
- The word 8 stands at disposal as status for every variable. The latest edited word number is registered here in higher valued byte. The number of bytes which were last written stand in the lower valued byte. This word can for example be zeroed by the PLC and in conclusion monitored on ><0, in order to wait for an input of the PCS operator.
- With bit variables, except for the edited ones, they all remain free for alteration. The respective altered bit is additionally registered in word 9 with logic 1. In this manner it can be found out which bit within the registered bits in word 8 has changed. The new state of the bits can be registered through and together with the bit mask as registered in word 9 as well as the amended word number which is registered in word 8.

2.4.1.6.5 ARROW KEYS IN MENUS

The arrow keys are permitted in a node to direct to further nodes as at the same time to further variables. If a non valid arrow key is pressed, an acoustic warning signals rings an error. A valid arrow key with LED is displayed, so long as bit 5 of word 13 is logic 0. Should an LED illuminate statically, then an additional variable can be selected from the same page. If on the other hand an LED flashes, then you can leave this node. These LED functions are only for the PCS 090, PCS 095, PCS 095.1.

If several nominal value variables are applied in a text, then these can be reached via the arrow keys. If there are several nodes in the activated menu, the arrow keys have a double significance (variables-, node exchange). If this is not required then only one nominal value variable per node or only one node per menu may be declared.

"ARROW-LEFT" "-RIGHT": If several nominal value variables are used in the text, then every line of a display page will be considered as lying side by side and the next variable will be looked for. In the case the arrow key LEDs are released and further nominal value variables to be edited are present, then the relevant LED lights up statically. If the actual variable had just been the last or the first, then the next node is looked for. If this is available, then position will be takten on the 1st variable top left. When the arrow key LEDs are active and a sequence node is available in the direction of the arrow, then this LED flashes. If there is no node in the direction of the arrow, an acoustic error signals rings.

"CURSOR UP" "-DOWN": If variables are distributed on several display lines, then the first variable in that line will be selected (left) which is in the direction of the arrow. If the arrow key LEDs active, then the corresponding LED will light up. If there is no nominal value variable in this line, then the next node in the direction of the arrow will be looked for. In so far as the arrow key LEDs are free, the relevant LED will flash in this case. If there is no node there, then pressing this key will sound an acoustic warning.

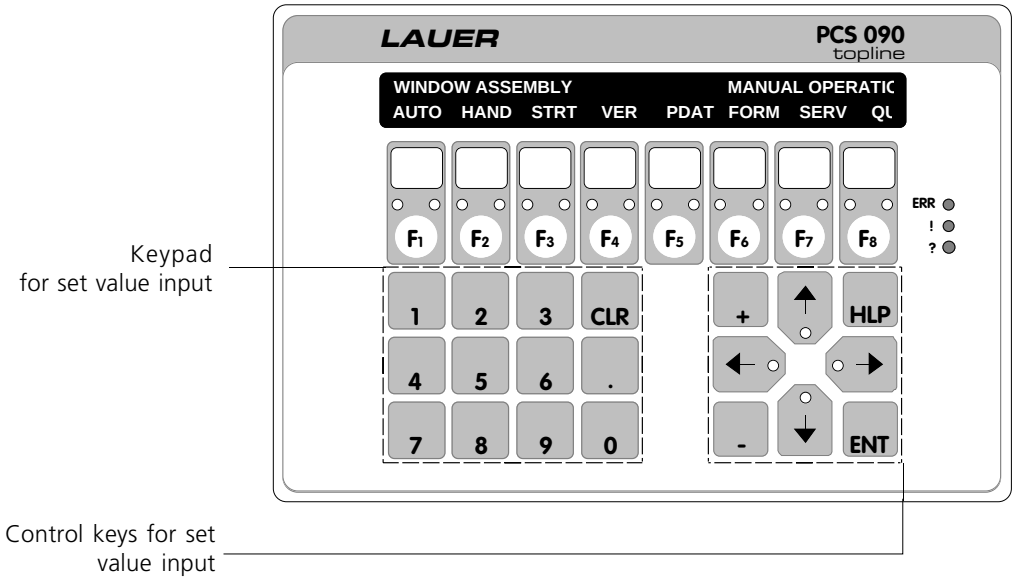
"CURSOR UP" in the last line and "CURSOR DOWN" in the first line always look for the next node.

Termination of a menu can be recognized in word 6, bit 0. If the bit is logic 0, the menu is no longer active. The exact point in time of the termination can be found through the negative edge triggering.

2.4.1.6.6 PERMISSIBLE KEYS IN MENUS

Operation of the integrated editors		
Variable Type	Key	Function
BIT	PLUS	A bit that was logic 0 sets to logic 1 (at once written in the PLC).
	MINUS	Deletes a bit that was logic 1 to logic 0 (at once written in the PLC).
	* ARROWS	Leaves this variable if allowed. The next variable or node in arrow direction is looked for.
STRING	* PLUS	Increments the value of a variable, so long as value is still within limiting values.
	* MINUS	Decrements the value of a variable, as long as value is still within limiting values.
	CLR	Restore the old value in the display; (the last value read by the PLC).
	ENTER	Writes the selected value in the PLC as long as it has been amended and not yet written.
	* ARROWS	Write the selected value, if it has been modified and not yet sent and then look for the next variables, i.e., the next menu nodes in direction of arrow.
CSTRING	* PLUS	Increments the value of a variable, so long as the value is still within the limiting value (in contrast to STRING at once stored in the PLC).
	* MINUS	Decrements the value of a variable, so long as the value is still within the limiting value (in contrast to STRING at once stored in the PLC).
	CLR	Restore the old value into the display; (the last read value of the PLC).
	* ARROWS	Leave these variables if allowed to. The next variable or node in arrow direction looked for.

* = Auto repeat



Variable Type	Key	Function
BCD-1 BCD-2 BCD0-1 BCD0-2	* PLUS/MINUS	Adds/subtracts n within the limiting values (Offset input), whereby * n = 1 if no numeral input follows, i.e., * n = given value, if numerical input follows.
	CLR	Restores old value in the display; (last read value by PLC)
	ENTER	Writes the selected value in the PLC, if it has been changed and not yet written.
	* ARROWS	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next node in the arrow direction.
	* NUMBERS	Permit direct input.
BIN-A BIN-B BIN0-A BIN0-B	* PLUS/MINUS	Adds/subtracts n within the limiting values (Offset input), whereby * n = 1 if no numeral input has occured, i.e., * n = value input, when numerical input has just been entered.
	CLR	Restores the old value in the display; (last value read by PLC).
	ENTER	Writes the selected value in the PLC, if it has been changed and not yet written.
	* ARROWS	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction.
	* NUMBERS	Enables direct input. Numbers moved from right to left (even beyond a decimal point).
BIN-1 BIN-2 BIN0-1 BIN0-2	* PLUS/MINUS	Adds/subtracts n within the limiting values (Offset input), whereby * n = 1 if no numerical input follows, i.e., * n = given value, if numerical input follows.
	CLR	Restores the old value in the display; (last value read by PLC).
	ENTER	Writes the selected value in the PLC, if it has been changed and not yet written.
	* ARROWS	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction.
	* NUMBERS	Enables direct input. Numbers are entered according to the pocket calculator principle.
	(*) POINT	Changes to after decimal point position, if after decimal point positions are defined.

* = Auto repeat; (*) = Auto repeat, though without significant meaning

Variable Type	Key	Function
VBIN-A VBIN-B VBIN0-A VBIN0-B	* PLUS * MINUS CLR ENTER * ARROWS * NUMBERS	Gives the operational sign "+". Gives the operational sign "-". Restores old value in the display; (last read value by PLC). Writes the selected value in the PLC, if it has been changed and not yet written. Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next node in the arrow direction. Permit direct input. Numbers moved from right to left (even beyond a decimal point).
VBIN-1 VBIN-2 VBIN0-1 VBIN0-2	* PLUS * MINUS CLR ENTER * ARROWS * NUMBERS (*) POINT	Gives the operational sign "+". Gives the operational sign "-". Restores the old value in the display; (last value read by PLC). Writes the selected value in the PLC, if it has been changed and not yet written. Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next node in arrow direction. Enables direct input. Numbers entered according to the pocket calculator principle. Changes to after decimal point positions, if after decimal point positions are defined.
WORD	* PLUS * MINUS CLR ENTER * ARROWS * NUMBERS	Moves the cursor one bit position to the right in direction of lowest value bit LSB. Moves the cursor one bit position to the left in the direction of the highest bit MSB. Restores the old value in the display; (last read value by PLC). Writes the selected value in the PLC, if it has been changed and not yet written. Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction. Only the keys <0> and <1> are significant: <0> Sets a bit to 0 and moves the cursor, if possible one position tho the right. If the cursor finds itself at end of the variables, then it will be positioned on the highest bit (MSB). <1> Sets a bit to 1 and moves the cursor, if possible one position to the right. If the cursor finds itself at end of the variables, then it will be positioned on the highest bit (MSB).

* = Auto repeat; (*) = Auto repeat, though without significance

Variable Type	Key	Function
ASCII	* PLUS	Presents the character with the next higher displayable character code. If the end of the character table has been reached, the first presentable character is returned.
	* MINUS	Present the character with the next smaller displayable character code. If the begining of the character table has been reached, the last character out of the character table is returned.
	CLR	Restores the old value in the display; (last value read by PLC).
	ENTER	Writes the selected value in the PLC, if it has been changed and not yet written.
	* CURSOR	Write the selected value, if it has been changed and not yet written and seeks the next variable, i.e., the next menu node in arrow direction.
	* POINT	Moves the cursor one place to the right. If the variable end has been reached, then the cursor will again be set on the first position of the variables.
WORD-KH	* PLUS	Increments the digit by 1 the cursor is positioned on.
	* MINUS	Decrements the digit by 1 the cursor is positioned on.
	CLR	Restores the previous displayed value.
	ENTER	Writes the selected value into the PCS, if it has been modified, but not sent yet.
	* ARROWS	Write the selected value, if it has been modified, but not sent yet and search for the next variable in the direction of the arrow.
	* NUMBERS	Used to directly enter numbers (0..9) on the corresponding digit.
	* POINT	Moves the cursor from left to right. If the rightmost cursor position is reached, the cursor is repositioned on the left digit.
WORD-KY	* PLUS	Increments the digit by 1 the cursor is positioned on.
	* MINUS	Decrements the digit by 1 the cursor is positioned on.
	CLR	Restores the previous displayed value.
	ENTER	Writes the selected value into the PCS, if it has been modified, but not sent yet.
	* ARROWS	Write the selected value, if it has been modified, but not sent yet and search for the next variable in the direction of the arrow.
	* NUMBERS	Used for direct entry: Numbers are shifted from right to left (calculator entry).
	* POINT	Switches between high byte and low byte of the data word (decimal format).

* = Auto repeat

2.4.2 ADMINISTRATION OF PRIORITIES

Several of the priorities in the PCS 009, PCS 090, PCS 095, PCS 095.1 can be active. It is always the highest active released priority that will be displayed. If a priority is deleted or barred, then the next lower active released priority will be displayed.

The behaviour of the PCS is shown by the PCS status, which is put at disposal in the PLC transfer area in the words 6...9, as well as in the command word A (W13):

- Word 6 (bit 0...5) displays every active priority, even when they are barred and therefore not on display.
- Word 7 (bit 8...11) shows the priority currently on display. This is the highest active released priority.
- In the command word A (W13; bit 8...11) several priorities (menu, information, warning and fault) can be barred at any time. This can be used, for example, to prevent a menu from being interrupted by a information, warning or fault as long as it is active.

Here is an example that will clarify the matter:

Assume that:

- Fault priority is barred (that means bit 11 of word 13 logic 0), every other one active (bit 8...10 of word 13 logic 1): W13, bit 8...11: 0111.
- The following priorities are active: Fault, information, menu: word 6 bit 0...5: 0x1011.

Valid therefore is:

- The highest presentable display, i.e., active priority is information (word 7, bit 11...8: 0100).

If the information is deleted, then the menu is the highest valued active released priority:

- Word 7, bit 8...11: 0010; word 6, bit 0...5: 0x1001.

If the operator presses the HLP key, and assuming that a help text has been programmed into the priority menu, then the help text will be displayed on account of its higher (not barred) priority. If the operator releases the HLP key, the menu will again be displayed.

As soon as the PLC again releases the fault priority (bit 11 of word 13 is logic 1), the fault message will be displayed.

If the fault message is then deleted, the menu will again appear.

When the operator terminates the menu after having controlled the PLC, the preselected default text will be displayed.

The priorities 0 to 8 allow themselves to be limited by the PLC (from the highest to the lowest).

Here are the individual priority classes:

Lowest	0	=	DEFAULT TEXT (operating texts 0...127) displayed when there is no higher priority switched on.
: *)	2	=	MENU (operating texts 0...127) activated and concluded through the PLC (word 14)
: *)	4	=	INFORMATION (message texts 0...127) activated through 0 > 1 transfers at least one message bit to which a text with INFORMATION priority has been allocated to. Deactivated according to the cancel mode of the corresponding INFORMATION message texts.
: *)	6	=	WARNING (message texts 0...127) activated through 0 > 1 transfers at least one message bit to which a text with WARNING priority has been allocated to. Deactivated according to the cancel mode of the corresponding WARNING message texts.
: *)	8	=	FAULTS (message texts 0...127) activated through 0 > 1 transfers at least one message bit to which a text with FAULTS priority has been allocated to. Deactivated according to the cancel mode of the corresponding FAULTS message texts.
:	12	=	HELP (Help texts on (R) default-, (M) enu-, (H) information-, (W) arnings- and (S) faults priority) activated by pressing <HLP> deactivated by releasing <HLP> Prerequisite: HELP-text is allocated to the corresponding priority.
: **)			ERROR PRIORITY (firm text) activated through interfaces or start test error usually deactivated through PLC RESET command.
highest			PLC stop/run - transmission or new start

*) These priorities are not activated, if they are barred through the PLC.

**) This error case will be caused mainly by driver in the error word W3 of the PLC. The executions of the error word are specifically to do with the driver and are therefore to be found in the manual PCS 91.xxx.

2.4.2.1 DEFAULT TEXT PRIORITY

The operating texts 0...127 belong to this priority class. They can all be applied as default texts. The operating texts can and will be used in menus. The PLC alone decides which of these default texts (bit 8...14 in word 14) are to be displayed and whether or not the default text should be flashing (bit 15 in word 14). The character and control keys do not have any function here. However if they are pressed, then the acoustic error will be suppressed in order that the control keys can be applied for control purposes. An exception here is the HLP key, which brings the defined help text in the default priority onto the display. Every variable can use NOMINAL values, NOMINAL-P-values and ACTUAL values. None the less nominal values can not be changed. Every variable is cyclically refreshed.

The DEFAULT TEXT Nr. 0 possesses a special position: it immediately appears after switching on the PCS even when no communication has been started with the PLC. Should a variable be in the idle text 0, then this variable will be replaced by an empty space until the variable out of the PLC can be read. This is an good way to recognize whether the communication has been started.

If an default text is selected that is not declared, then the previous displayed default text will remain active.

2.4.2.2 MENU PRIORITY

127 menus are available for this priority class (2). The menus are labelled with numbers from 1...127. A menu consists of one or more nodes (1...255), whereby an operating text (0...127) is allocated to every node.

A menu is called with the command word B (W14), bit 0...6.

Requirements for starting up the menu priority is that a menu is programmed and that no higher priority prevents the start up of the menu.

The actual node number is displayed in the word 6, bit 8...15 as status.

Within a menu additional nodes can be reached with the arrow keys, whereby the structure is freely programmable. The first declared node is the initial node or start node. This initial node is activated by the call of the menu.

It will be determined through bit 7 in word 14, whether nominal-P-variables can be modified or not. If bit 7 logic 0, then only nominal values can be modified. If bit 7 is logic 1, then nominal value variable and nominal-P-variable can be modified. This bit can be changed by the PLC at any time, e.g., node dependent.

2.4.2.3 MESSAGE PRIORITIES

In these priority classes (4, 6 and 8) texts are called by the setting of a bit in the message area word 15 up to maximum word 22. A message text with maximum 32 lines is allocated to each of the 128 bits. An individual MESSAGE PRIORITY can be determined for each of the 128 texts (fixed through programming). Here they are individually.

- **INFORMATION PRIORITY (Priority 4)**
- **WARNING PRIORITY (Priority 6)**
- **FAULT PRIORITY (Priority 8)**

These priority classes differenciate themselves only on the priority level and not in there function. For every priority class there is an individual storage behaviour (word 12 bit 0...5) and an individual display behaviour (word 12 bit 8...10) which is controlled by the PLC (and therefore can be changed over at any time). Refer to the following section for further information on this matter.

Should a message bit be set, to which no message text has been declared, then there will be no reaction.

2.4.2.3.1 STORAGE BEHAVIOUR

- **FIRST VALUE MESSAGE WITHOUT THE POSSIBILITY OF MANUAL SELECTION:**
  The oldest message text remains in the display until it is deleted.
- **FIRST VALUE MESSAGE WITH THE POSSIBILITY OF MANUAL SELECTION:**
  The first bit that has a positive edge (0 > 1 transmission) brings its text in the display. If additional bits are set, then these texts can be reached with the key "ARROW-RIGHT". Reverse is achieved with "ARROW-LEFT". The text inputs can at any time and randomly (depending on their cancel modes) be deleted from the memory. If the bit 14 in word 13 is logic 0, then the manual selectibility, in so far as more than one message is active, will also be displayed with the arrow key LEDs (left and right).
- **LAST VALUE MESSAGE WITHOUT THE POSSIBILITY OF MANUAL SELECTION:**
  Every 0 > 1 transmission brings its text immediately into the display, the older inputs remain in the storage. In case the most recent message text is deleted, the next most recent message text will appear in the display.
- **CYCLICAL DISPLAY WITHOUT THE POSSIBILITY OF MANUAL SELECTION:**
  This kind of storage is in accordance with the first value message. If however several texts are switched on, then the inputs cycle with a programmable scroll time. If further switching is made to the help texts, then the scroll time will be started from new (stopping time = 0,5 s). In principle all cancel possibilities are also possible here. Nevertheless as there is no prohibitive time for keys within a priority, only cancel possibility 1 (no manual deletion) should be chosen (in order to avoid operation errors).

For example: The cyclic display is activated. At the moment there are more than two messages of the same priority active. Every message is programmed with deletion characteristics 2 (therefore can be deleted manually). The operator ascertains that the currently displayed message can be quitted and he/she presses the CLR key. At the same time, for example, the cyclical display switches to the next message, the false message is now quitted!

The written storage behaviour is at any time individually adjustable with the bits 0...5 in word 12 for every priority (information, warning and fault). In this way it is also possible for example to amend the storage behaviour by means of a priority change. Changing the storage behaviour only influences the display characteristics and not the input behaviour. In order to avoid incorrect operation, a blocking time of 0.5 seconds for the controlling keys is built in after a priority changeover.

Basically an attempt is being made to also register the chronological appearance of the flags into the correct chronological sequence. The following limitations are however also imposed. The reading off of the bits has a relatively low priority in comparison with the other tasks of the PCS.

Should several bits be set in a cycle, the lower text numbers have a higher priority.

2.4.2.3.2 CANCEL MODES

The cancel mode is individually programmable for every message bit. It is determined by programming with the programming software PCS*PRO* . There are 4 kinds of deletion:

Cancel mode 1 or deletion through the PLC:
The text remains switched on as long as the relevant bit is = 1. If the PLC resets the bit, then the message text will be cancelled. The bit is merely read by the PCS. The operation required LED (?) is switched off.

Cancel mode 2 or manual deletion by resetting the message bit:
The text is switched on with an 0 > 1 transfer and can be quitted with <CLR>. Thereby the message bit is cancelled in the PLC and as a result of the deleted message bit, the text is switched off. Reversing the message bit, on the part of the PLC, has the same effect as pressing the <CLR> key. The message bit in the PLC programme may only be set once in order to obtain this deletion behaviour (no current assignment "!"), since the message would again be displayed after the <CLR>.
After activating the <CLR>, the operation requirements LED (?) immediately goes out.

Cancel mode 3 or manual deletion without reversing the message bit:
The text is switched on with an 0 > 1 transfer. Switching off of the text is possible at any time with the <CLR> key independent of the status of the message bit. The message bit itself (in the PLC) must be reversed with the PLC programme.
After activating the <CLR>, the operation requirements of the LED (?) goes out immediately.

Cancel mode 4 or manual deletion, if the message bit is 0:
The text is switched on with every 0 > 1 transfer. The text can be switched off with the <CLR>, when the PLC sets the message bit to 0. The state of the message bit is displayed through the operational requirements-LED (?).
Flashing: The bit is still log 1, deletions not possible.
Continual light: The bit is log 0, the message may be deleted.

2.4.2.3.3 DISPLAY BEHAVIOUR

An individual display mode can be controlled at any time by the PLC for any of the priorities 4, 6 and 8, also INFORMATION (bit 8 of word 12), WARNING (bit 9 of word 12) and FAULT (bit 10 of word 12). Through evaluating word 7 (displayed text number), for example, can be defined dependent of message texts.

There are two display characteristics:

The bit is logic 0: the message text is static
The bit is logic 1: the message text flashes

2.4.2.3.4 VARIABLES IN MESSAGE TEXTS

Basically every variable within INFORMATION and WARNING priority are treated as actual values. They are continually refreshed. Editing of the variables is not possible.

2.4.2.4 HELP PRIORITY

This is the highest priority level, which in normal circumstances is always attainable by the user. It is active as long as the "HLP" key is pressed. If it is released, the priority will switch off again. This priority step is not effected by the PLC and is available to the user at any time as long as a HLP text has been assigned to the currently displayed priority in the display. The text required here is an independant text with a maximum 32 lines. It is advisable to integrate every variable, in case an eventual error diagnosis could be of significance (even internal variables!).

Individual help texts are available for the respective priorities 0...8 (default, menu, information, warning and fault).

If a switch is made to the following lines by activating the HLP key with ARROW DOWN and then releasing the HLP key, the line numbers remain stored (retentive). On activating the HLP key again, the previous operating text page comes back into the display. With ARROW UPWARDS (on pressing the HLP key) the main lines can be brought back.

The arrow key LEDs show, as long as bit 15 of word 13 is logic 0, to the operator whether or not the following lines or the main lines can be reached.

The help text for the default priority is only available for continual refreshing. In other cases (only after RESET voltage off/voltage on) fixed texts will be displayed which serve diagnosis. Refer to the chapter "DIAGNOSTIC TEST".

2.4.2.5 ERROR PRIORITY

The highest priority level described here is activated through various errors. The respective texts can not be modified. These texts exist in the G version as English abbreviations:

The following errors can only appear when switching on (self tests):

```
====== INTERNAL ERROR ======
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

There appears for XXXX in the second line:

- **INVALID CHECK SUM IN DATA MEMORY:**
  An invalid data record is included in the flash EEPROM as the check sum is not correct.
- **NO PLC DRIVER FOUND:**
  The driver which is necessary for the communication with the PLC is not present.
- **UNUSABLE DIRECTORY:**
  The logical data structure in the EEPROM is not correct.
- **INITIALIZATION FAILED:**
  On running through the initialization programme of a driver or function, an error occurs for one or other reason. Have you loaded the correct driver?
- **FIRMWARE ARE NOT COMPATIBLE:**
  A data record, function or driver is loaded which is not compatible with the EPROM version. For example, some PLC drivers require a specific EPROM version.
- **DRIVER IS NOT SUPPORTED:**
  A data record, function or driver is loaded which cannot run with this hardware. Check out the hardware in use.

Basically the data record should once again be thoroughly checked out after one of these failures has appeared. After transmission, for safety reasons, the DIL switch 9 should certainly be switched to "off". If this error occurs again, then it is the result of a hardware fault.

As well as these stored error messages, there are error messages of the loadable driver and functions programme. Details of which can be found in the appendix. One error message is however common to all drivers:

```
===== COMMUNICATION ERROR =====
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

This is a message of the communication driver which conveys whether the communication connection to the PLC was interrupted or badly corrupted. The following line is reserved for the execution of the special driver. See the special manual PCS 91.xxx.

Warning!
After a communication loss, all actions which should have been performed by the programmable controller during the communication interruption are transmitted by the PCS to the programmable controller. The correct action/ reaction of the programmable controller and the PCS have to be checked after a restart!

2.4.2.6 DIAGNOSTIC TEXT

After successful initialization (no "internal error" of the PCS 009, PCS 090, PCS 095, PCS 095.1), a diagnostic test can be called without the communication running to the PLC (LED-"ERR" displays a continual light).

This happens by pressing the HLP key. It is possible to page forward after pressing the ARROW DOWN key onto the additional lines. Application of the ARROW UP key again presents the main lines. Releasing the HLP key, the display is suspendend and will be again be actuated by pressing the key. This procedure is however not retentive.

The following diagnostic information, which you should have to hand if possible for **telephone enquiries** can be read as follows:

- Equipment identification and version number of the EPROMS
- Information on the data record DAT: name of data record, number of least version EPROM, date and time of the creation of the data record and software name, with which the texts are made (PPCS or PCSPROX.X).
- Information on the driver DRV: Project driver name with date and time, number version EPROM, with which the driver runs, driver version and every available driver variation with actual settings.

  Especially the driver variables "AC...AF", mainly occupied with baud rates and interface types, inform on the possible interface settings. The currently position of the DIL switch 5, 6 is marked at the beginning of the line with the arrow ">".
- Information on the functions (should these be available) FKT: Project function name with date and time, original function time with date and time, the least number of the version EPROM, with which the function runs, function versions and every available function variables with actual settings.

3 ACTIVATION OF THE PCS 009, PCS 090, PCS 095, PCS 095.1

Activation of the PCS 009, PCS 090, PCS 095, PCS 095.1 is done over a transmission area which has a maximum number of 256 words and lies in the PLC.

In this manner the operator controls every function of the PCS. With "writing" and "reading" he/she controls this data.

Activities which lead to the communication between the PLC and the PCS are organized by the accompanying PLC software, which is in the respective manual PCS 91.xxx. Data security and communication records are taken care of by the communication's processor (e.g., PCS 810.1), i.e., the operating system of the PLC and the PCS.

As the transfer area (e.g., its position, size and functionality) is dependent on the parameterized driver, the corresponding **"Driver manual" PCS 91.xxx** should be used as well.

Basically **two principles of the data exchange** are available:

First principle An „expander" enabling all functions of the PCS is required for the PLC. Only the size of the transfer area varies depending on the driver. At any time It is possible (except for interrupt programs) to access data of the transfer area. It is the task of the expander to process the coded data packages of the PCS. Only the currently required data and task are transmitted via the interface.

Second principle „Direct drive" (e.g. „AS511.DRV"). It has the advantage that it reads and writes data from/into fixed data areas in the PLC. This principle is used for systems having no commands for indirect addressing or which require a relatively large amount of PLC cycle time for processing the expander. Below, you will find a brief description of the data exchange.

The PCS writes data into a data area of constant length in the receiving area of the PLC. At a later point in time the PCS reads data from a data area of constant length in the sending area of the PLC. Please note that the sending and receiving area are located in different data areas.

Words written by the PCS are stored In the receiving area of the PLC. Examples for the written data are key words, PCS status information, and external variables (set values).

Words read by the PCS are located in the sending area of the PLC. Examples for the read data are LED status, command words, message area, and external variables (set and actual values!).

Special attention must be paid to the reading and writing of data from/into the sending/receiving areas (especially for set values). Since the PCS reads and compares the set value twice after writing, the application program must transfer the written data from the receiving area of the PLC into the sending area. This can be only be done at a certain time slot since only then all data of the receiving area are (consistently) valid. For this purpose, a special block (subroutine) is available in the handling software of the PLC which is timely executed between writing into the receiving area of the PLC and reading from the sending area of the PLC. Only during this time slot, data of the sending/receiving area should be accessed. Especially set values must be reflected during this time. If access to these words must be possible at any time, then the data must be copied into/from temporary flag(s).

With this method, the functionally and the number of variables is reduced in comparison to the first principle. Deletion behavior 2 for messages and modifying of a bit in the transfer area cannot be realized. This applies only for using the unsynchronized operation.

The remaining chapter 3 describes the operation independent of the programmable controller. It is to be supposed that the full functionality is available. The transfer area is consecutively numbered from word 0 to word 255 - in short **W0 .. W255**.

Warning!
Take care to use the appropriate driver for the programmable controller. Otherwise, malfunctions can be caused in the PCS and in the programmable controller!

1. System area: W0...3

W0...2 used internally, barred from the user.
W3 Error word for the communication display. Details in the respective "Driver Manual" PCS 91.xxx.

2. Fixed function area: W4...14

		Direction: PCS PLC
KEYS:		
W4	Key bits F1...F8, DIL 1...4, HLP, ., 8, 9	———>
W5	Key bits 0...7, ARROWS, +, -, CLR, ENT	———>
PCS STATUS:		PCS PLC
W6	Node number and active priority	———>
W7	Displayed priority and text number	———>
W8	Word number nominal value and nominal value length (byte number)	———>
W9	Bit mask high and low (only for bit variables)	———>
LED STATUS, DISPLAY AND STORAGE MODES:		PCS PLC
W10	LED activation F1...F8 green and yellow	<———
W11	LED flashing status F1...F8 green and yellow	<———
W12	Display and message mode (storage behaviour)	<———
COMMAND WORDS:		PCS PLC
W13	Release of priorities and transmission	<———
W14	Default text number and menu number	<———

3. Message area: W15...22

	PCS PLC
Cancel mode 1	<———
Cancel mode 2	<———>
Cancel mode 3	<———
Cancel mode 4	<———

4. Extension area: W 23...29

	PCS PLC
W23.. 29 currently used for variables in the PCS 009 and PCS 090	<———>

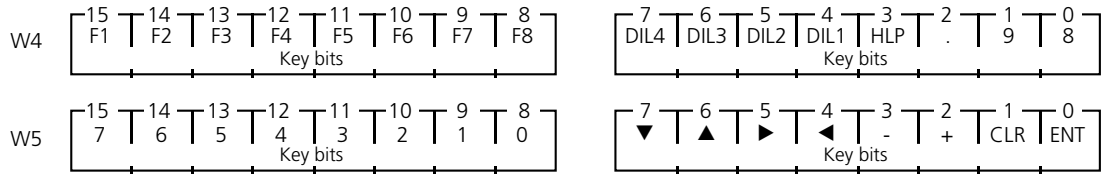
		PCS PLC
ADDITIONAL KEYS (ONLY PCS 095, PCS 095.1):		
W23	key bits F9...F16 in high byte (only PCS 095, PCS 095.1)	———>
ADDITIONAL LED STATUS (ONLY PCS 095, PCS 095.1):		
W24	LED activation F9...F16 green and yellow (only PCS 095, PCS 095.1)	<———
W25	LED flashing status F9...F16 green and yellow (only PCS 095, PCS 095.1)	<———
W26...29	currently available for variables	<———>

5. Area of variables: W30...255

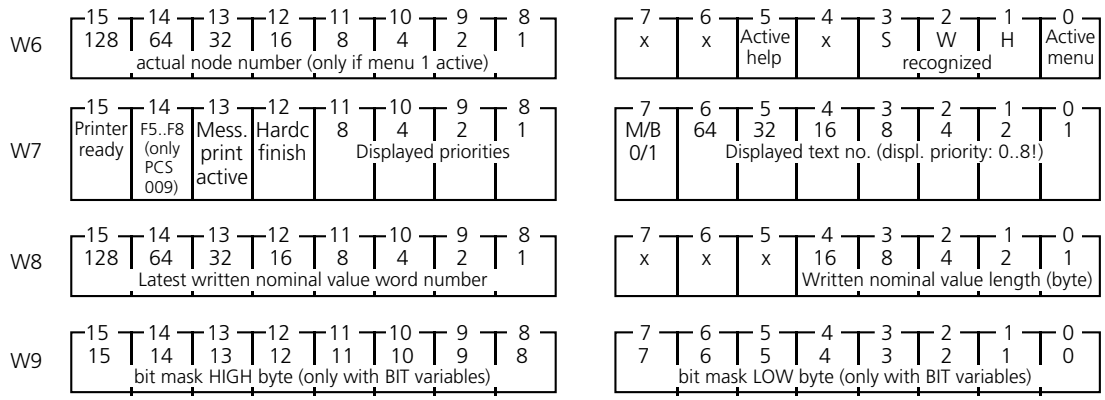
	PCS PLC
W30...255 free for variables	<———>

3.1 SHORT SURVEY OF THE TRANSFER AREAS

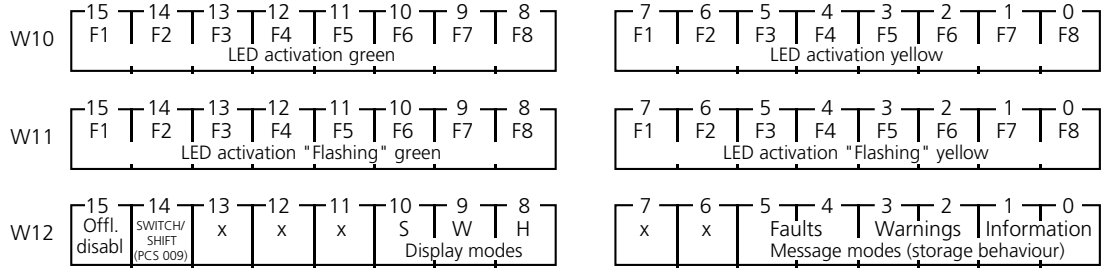
KEYS:



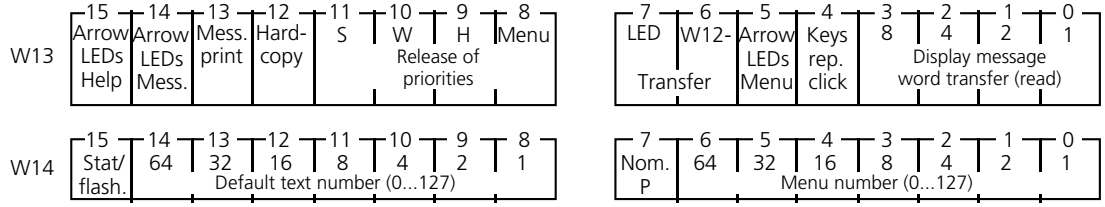
PCS STATUS:



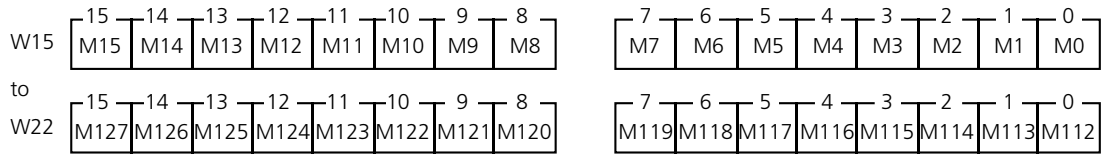
LED STATUS, DISPLAY- AND STORAGE MODES:



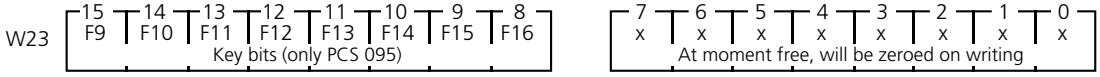
COMMAND WORDS:



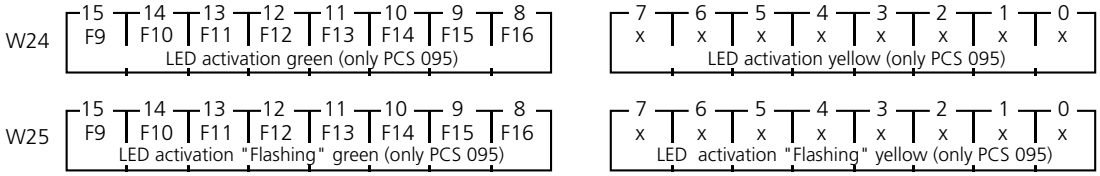
MESSAGE WORD:



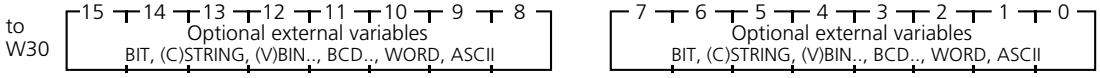
EXTRA KEYS WORD: (ONLY PCS 095, PCS 095.1)



EXTRA LED STATUS: (ONLY PCS 095, PCS 095.1)

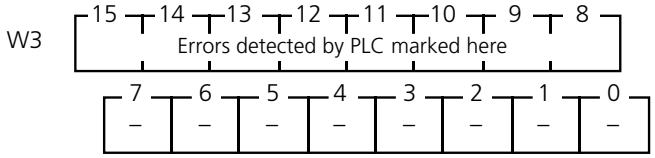


AREA OF VARIABLES:



3.2 SYSTEM AREA

The words W0...2 are reserved for driver dependent functions.



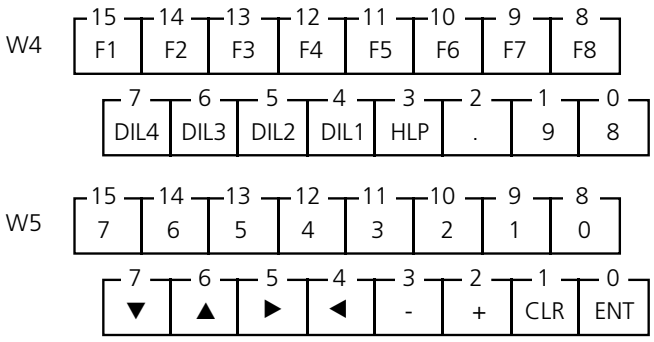
Errors detected by the PLC are marked in the HIGH byte of word 3. As these errors are driver dependent, is useful to refer to the manual PCS 91.xxx. If an error occurs the cause of it can be read. This information is to be on hand especially with **telephone enquiries**

This error concludes the termination (PLC - possible on a page!). First after a communication RESET with the PLC is an attempt made to start the communication again. A return is made to the last active position of the PCS. This is predetermined by a run procedure of the customer orientated previous occupied and emergency programme in the PLC, information can be lost. However there is a particular programming which could reverse this situation.

3.3 FUNCTION AREA

The key bits as well as the PCS status, will be read here from the viewpoint of the PLC operator. The LEDs, display and storage modes and the command words A, B are here written.

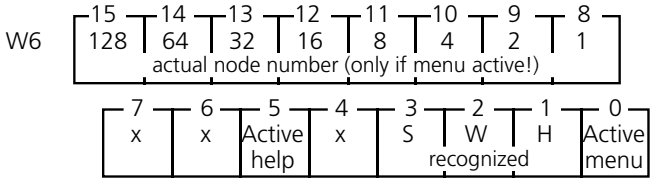
3.3.1 FUNCTION, CONTROL AND NUMERIC KEYS



These key bits are log 1, so long as the respective key is pressed and the communication runs without fault. The control keys in the LOW byte of W5 should only be used with care, as they are also required for the editing of nominal values in several priorities, especially in the menu priority for editing.

Mint: The ENT key should not be used for the termination of menus, as it is also used for the transfer of nominal values. A function key as menu end key is better suited for this purpose. Termination of a menu occurs simply by writing logic 0 on bit 0...6 of word W14. The final menu node can be checked through the PCS status W6 bit 0 (negative edge).

3.3.2 PCS STATUS



HIGH byte, bit

0...7: actual node number (binary)

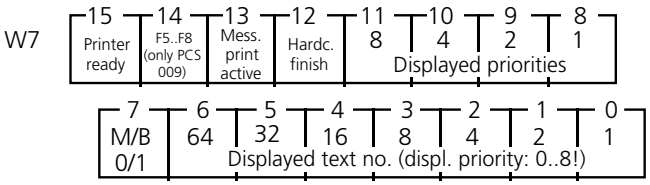
The actual node number (1...255) is only active as long as a menu is active. Whether a menu is active or not can be read in the same word in bit 0.

LOW byte, bit

6...7: not used at present
5: log 1, as long as a Help text is displayed
4: not used at present
3: log 1, if at least one fault is active
2: log 1, if at least one warning is active
1: log 1, if at least one information is active
0: log 1, if one menu is active

If the HLP key is pressed by the operator, the programmed help text appropriate for the currently displayed priority is set for the bit 5, as long as the help key is pressed. If there is a message bit log 1 in the message area W15...22, and at the same time a message text is programmed, then the (H) information, (W) arning, or (S) fault respective to the programmed priority of the message, will be recognized. A prerequisite for this is however that **the message word transfer is free**, refer to LOW byte of command word A (W13)!
If a menu was started with the command word B (word 14; bit 0...6), that is filed in the data record, then the bit will be log 1. This bit can also be brought in as end criterion (negative edge) for a menu.
Several priorities can be active and therefore several bits can set. The presently displayed priority can be evaluated in the HIGH byte of word 7.

Example:
A message (fault) is active, however barred through the bit 11 in command word A (W13) (not displayed). At present menu 15 and the node number 28 is in the display. In addition the HLP key is not depressed. Content of word 6: 00011100 xx0x1001.



HIGH byte, bit

12	logic. 1 if printing has been finished. This bit is kept at „1“ until bit 12 DW 13 „Request bit“ is set to „0“ (PCS 095.1 with printer interface only).
13	logic. 1 if message printing is active.
14	active F-key level. 0 corresponds to F0 .. F4, 1 corresponds to F5 .. F8 (only PCS 009).
15	logic. 1 if printer is ready for operation. Only with RTS/CTS handshaking, the „Ready“ printer state is clearly defined. With XON/XOFF handshaking, „Busy“ will only be signaled after receiving XOFF. Without handshaking „Ready“ will always be signaled (only PCS 095.1 with printer interface)
8..11:	priority currently used for display (binary, possible 0, 2, 4, 6, 8, 12)

The actual displayed priority is displayed as binary coded. This is the highest valued, free and active priority in command word A (W13). The following values are possible:

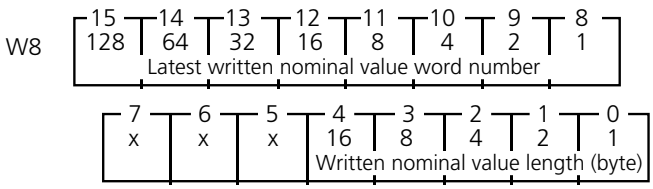
0	0000: Default priority
2	0010: Menu priority
4	0100: Information priority
6	0110: Warning priortiy
8	1000: Fault priority
12	1100: Help priority

LOW byte, bit

7:	log 0 if a message text and log 1 if an operating text is in the display.
0..6:	presently displayed text number (binary), only valid if the displayed priority is 0...8.

The currently displayed text number is shown here binary coded. It can be ascertained by checking of bit 7 whether a message text or an operating text is in the display. This bit is log 0 with message texts and log 1 with operating texts.

Example 1: Default text 3 is in the display -> word 7 (binary) = xxxx0000 10000011
Example 2: Operating text 34 (e.g. in the menu) -> word 7 (binary) = xxxx00010 101000010



HIGH byte, bit

0...7: latest written set value word number (binary)

The word number of the latest edited set value can be read in binary here. If an PLC programme is waiting on the input of a specific set value, W8 (or only the HIGH byte) can be zeroed before. As soon as the byte is <> 0 the set or nominal value can be evaluated in the PLC programme. If this doesn't correspond to the expected set value, then the W8 is to be zeroed again and waiting will continue.

LOW byte, bit

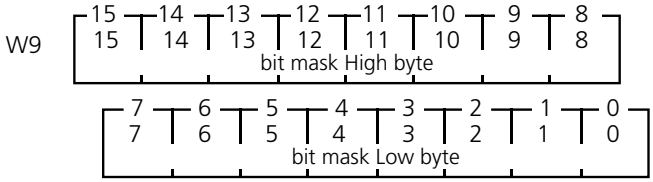
5...7: at present not in use
0...4: latest written set value length (binary, number of bytes)

If a set value is written by the PCS, the type of the variables on evaluation of the bits 0.4, the number of written bytes and those of the subsequent written bit masks can be evaluated.

Number of bytes:

0: BIT variable
2: 16 bit variable as (C)STRING; BCD(0)-1; (V)BIN(0)-1, A; WORD; ASCII
4: 32 bit variable as (C)STRING; BCD(0)-2; (V)BIN(0)-2, B; ASCII
[4: ASCII variable

If a bit variable (number of bytes = 0) is written, the amended bit number can be verified with the displayed bit mask in W9.



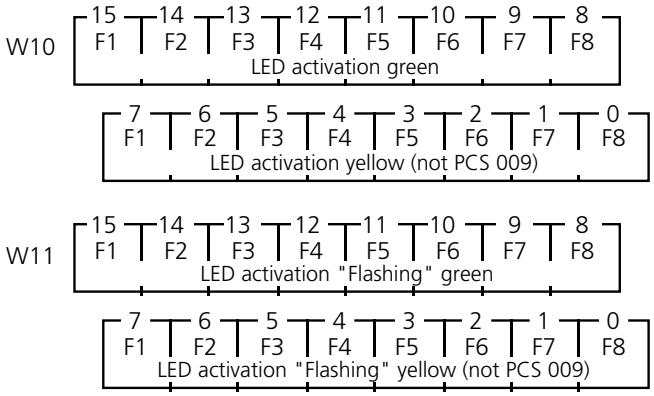
The bit number from the bit mask of W9 can be ascertained. The corresponding bit that has changed is reported in the bit mask with a logic 1. All remaining bits appear with logic 0. The new condition of the respective bits can be decided upon in conjunction with the word number entered in W8 and the bit mask.

Example:
A bit variable which was programmed on word 40 (bit 11) as a set value is adjusted in a menu (previously logic 0, after pressing "+" key logic 1). Afterwards the following values are present in W8 and W9:

W8: 00101000 xxx00000
W9: 00001000 00000000

Through a logical link up of the words W9 and W40, you have the values 00001000 000000, that is to say <> 0. Whereupon the bit is set to logic 1.

3.3.2 LED STATUS F1...F8



A green and a yellow LED (not for PCS 009)per function key F1...F8 is available. Every LED can take on 4 conditions, as each LED has 2 bits at its disposal:

- off
- on
- flashing (75% bright phase, 25% dark phase)
- inverse flashing (25% bright phase, 75% dark phase)

If an LED is flashing and another one is flashing inverse, then they light up alternately. A bright phase with an LED is a dark phase with another one and reversed.
The state of an LED is dependent on the 2 respective bits of word W10 and W11 (with the red LED via the function key "F6" these are e.g., bit 2 of W10 and W11).

Arrangement of the LED status:

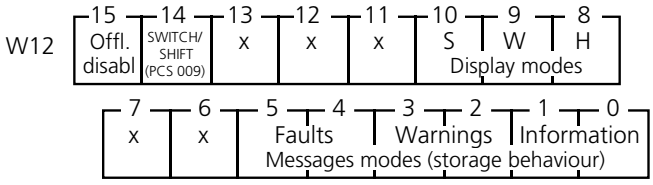
W10, bit no. x	W11, bit no. x	Condition
0	0	off
0	1	inverse flashing
1	1	on
1	1	flashing

Example:

W10: 00001111 00001111
W11: 00000000 11111111

-> green LEDs over F1...F4: off
-> green LEDs over F5...F8: on
-> yellow LEDs over F1...F4: inverse flashing (not for PCS 009)
-> yellow LEDs over F5...F8: flashing (not for PCS 009)

3.3.3 DISPLAY AND STORAGE BEHAVIOUR



HIGH byte, bit

15: logic. 1 if offline menu should be disabled
11..13: currently not used
14:*) Mode keys ▲/▼ logical 0 corresponds to the switch function F1..F4/F5..F8,
logical 1 corresponds to the shift function F1..F4/F5..F8
10: logic 1, if a displayed fault should be flashing
9: logic 1, if a displayed warning should be flashing
8: logic 1, if a displayed information should be flashing

DISPLAY MODES:

The display mode bits determine, whether a message text should be presented statically or flashing. A logic 1 means flashing presentation for the entire message text and a logic 0 represents a static presentation. These bits, separated for (H) information, (W) arning and (S) fault, can be altered at any time by the PLC.

LOW byte, bit

6...7: at present not in use
4...5: storage behaviour for faults
2...3: storage behaviour for warnings
0...1: storage behaviour for information

STORAGE BEHAVIOUR:

HL bit:

- 00: First message without selection possibility
- 01: First message with selection possibility
- 10: Last message without selection possibility
- 11: Cyclic display (without selection possibility)

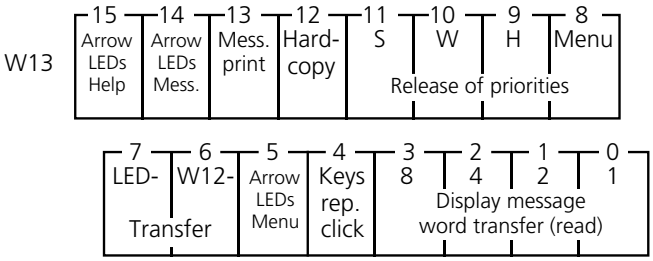
Continued switching to other messages by hand (with arrow right & left) is only possible in mode 01 (first message selection possibility). If paging is to continue in another mode, then mode 01 should be set first of all. At this moment the text currently visible in the display can be either paged with the "Arrow-left" to the oldest, i.e., "Arrow-right" to the latest message.

If the bit 14 logic 0 is in the command word A (W13), then the arrow key LEDs show in addition the paging possibility within the message text page and from the oldest to the youngest message (manual selection possibility).

The various cancel possibilities and the paging to the help lines are written in section "MESSAGE PRIORITIES".

*) only for PCS 009

3.3.4 COMMAND WORD A: RELEASE



HIGH byte, bit

15: logic 1, if arrow LEDs in the help priorities not to be active
14: logic 1, if arrow LEDs in the message priority (H, W, S) not to be active
13: logic. 1 if the message word area should be checked for 0 -> 1 transitions and to earmark messages for printing. Logic. 0 deletes all earmarked messages.
12: logic. 1 initiates the hardcopy function for the current display contents (only PCS 095.1 with printer interface).
11: logic 1, if the fault priority may be interrupted by a lower one (W, H, M, R)
10: logic 1, if the warning priority may be interrupted by a lower one (H, M, R)
9: logic 1, if the information priority may be interrupted by a lower one (M, R)
8: logic 1, if the menu priority may be interrupted by a lower one (R)

LOW byte, bit

7: logic 1, if the LEDs (W10...11, with PCS 095, PCS 095.1 also W24...25) must be read
6: logic 1, if the display and storage behaviour W12 is considered
5: logic 1, if the arrow LEDs should not be active in the menu priority
4: logic 1, if the acoustic signal (repeat click and error signal) should not be active
0..3: gives binary (0...8) the number of the message words, which have to be transfered (read)
1: logical 1 if at least 1 info is active
0: logical 1 if a menu is active

Interlocked Arrow Key LEDs: (only PCS 090, PCS 095, PCS 095.1)

The arrow key LEDs in the bits 15, 14 and 5 can be activated according to help, message (H, W, S) and menu priority. If the respective bit is on logic 0, the corresponding LEDs light up via the permitted "Arrow keys". More detailed information can be found in chapter "LIGHT DISPLAYS" and "ARROW KEYS IN MENUS". Here is a summary:

LED Arrow...	Menu priority	Message priority	Help priority
lit above	variable accessible	select main lines	select main lines
flashes above	menu node accessible	–	–
right lit up	variable accessible	latest report	–
right flashes	menu node accessible	–	–
lit below	variable accessible	select next lines	select next lines
flashes below	menu node accessible		–
left lit up	variable accessible	next older message	–
left flashes	menu node accessible		–

Release Priorities:

The priorities in the bits 8...11 (menu, information, warning, error) can be blocked individually. Please note that the priorities 8 (HELP) and 12 (communication errors) are not blockable. If on entering the set value into a menu, the priorities information, warning and fault are blocked, then only default texts and menus are allowed in the display (High byte = xxxx0001), the menus cannot interrupt the messages (H, W, S). Nevertheless they are activated in the background (LED "!" then flashes).

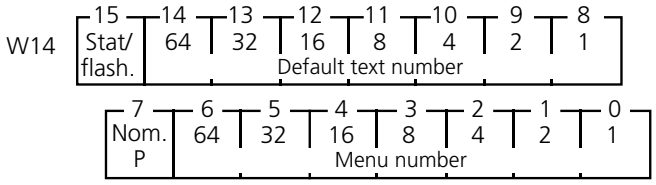
Release Data Transfer:

If certain words of the function area, such as LEDs, messages, displays and storage behaviour, are not required or do not have to be continually refreshed, then it is recommended that the transfer between PCS and PLC in the LOW byte should be kept to a minimum. The **blocking of the transfer of data saves time in the PLC cycle** and accelerates the transfer for other services such as data!
For example a logic 0 in bit 7 blocks the reading (transfer) of all the LED STATUS WORDS (W10...11 and with the PCS 095, PCS 095.1 also W24...25).
A logic 0 in bit 6 blocks the reading (transfer) of the display and storage behaviour. The number of message words in the bit 0...3, which should be read (transfered) can be set binary coded. Here, a value range between 0 and 8 is sensible.

Example: You require only 35 messages (35/16 = 2, 18...), therefore it is enough to be able to read 3 word messages. This can through writing xxxxxxxx xxxx0011 in W13. Please note that this setting can at any time (dynamic) be altered by the PLC.

Warning!
Blocking of the transfer can lead to problems, if it is carried out at the wrong moment. For example after blocking transfer (assuming deletion behaviour 4), of a set message, it cannot be deleted for the time being, even though the message in the PLC is on logic 0. Therefore only use the blocking facilities of the transfer when you are sure that this will not lead to any undesired consequences.

3.3.5 COMMAND WORD B: DEFAULT TEXT, MENU



HIGH byte, bit

15: logic 1, if a displayed default text should be flashing
8...14: binary coded the default text number (0...127)

Default Text Number:
This is the operator text number, when the default text priority is activated that will be displayed. By writing the value xxxxxxx xxxx0000 into W13 the default text priority is displayed. Only a filed help text or a communications error can interrupt the default text. The default text number which is identical with the operator text number, can be altered at any time by the PLC. The variables (set value) contained in the text are not adjustable (cannot be edited).
If bit 15 is logic 1, then the entire default text can be flashed. If the bit 15 is logic 0, then the default text appears static.
Example: Default text 23 is to be displayed. Therefore the value 1001011 x0000000 is to be written on the word W14.

LOW byte, bit

7: logic 1, if the NOM-P-variables in the menu priorities may be edited
0...6: the activated menu number as binary coded (1...127, as 0 = end of menu)

Menu number:
In writing a value on bit 0...5, you can activate a menu and end one. The requirement is that no higher priority (H, W, S, HLP or communications error) is active. By writing a binary coded value > 0, i.e., 1...127 on bit 0...6, a menu can be started. The first text to arrive in the display is the defined operating text for the initial node or start node. If there are already one or more nominal variables in a menu node, they will be activated first of all. They then appear flashing. Now set values can be entered and the menu, depending on the definitions which you have set in it, be branched out with the "Arrow keys". A valuable help to you at any time are the arrow key LEDs provided bit 5 of this word, logic 0.
If you want to end a menu, then simply zero the bits 0...6. In the case that a set value was just activated; indicated by flashing or with a cursor; this can still be written in the PLC within the permitted limiting values!
The NOM-P-variables in the actual displayed menu nodes can be altered at any time with bit 7 = logic 1. This enables the formation of keyswitch dependent menus or variables. NOM-P-variables are then treated as ACTUAL values. A very special case arises when a NOM-P-variable is being edited and the PLC during this time sets the bit 7 on logic 0. This then develops in such a way that this variable is then written in the PLC. If the present edited value or intermediate result happens to be outside the limiting values, whether minimum or maximum, then the PCS waits until a reliable value is entered. In conclusion the first NOM variable is searched for by the beginning of the text and presented in a flashing manner. If there is no NOM variable in this menu node, then all variables will be presented as ACTUAL values.
Released and vacant arrow key LEDs in menus, are able to show at any time and in dependent of the bit 7, the actual valid arrow keys.
Example: Menus 15 can be started at any time by writing xxxxxxx x0001111 on W14.

3.4 MESSAGE AREA

	15	14	13	12	11	10	9	8
W15	M.15	M.14	M.13	M.12	M.11	M.10	M.9	M.8
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.7	M.6	M.5	M.4	M.3	M.2	M.1	M.0
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
W16	M.31	M.30	M.29	M.28	M.27	M.26	M.25	M.24
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.23	M.22	M.21	M.20	M.19	M.18	M.17	M.16
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
W17	M.47	M.46	M.45	M.44	M.43	M.42	M.41	M.40
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.39	M.38	M.37	M.36	M.35	M.34	M.33	M.32
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
W18	M.63	M.62	M.61	M.60	M.59	M.58	M.57	M.56
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.55	M.54	M.53	M.52	M.51	M.50	M.49	M.48
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
W19	M.79	M.78	M.77	M.76	M.75	M.74	M.73	M.72
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.71	M.70	M.69	M.68	M.67	M.66	M.65	M.64
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
W20	M.95	M.94	M.93	M.92	M.91	M.90	M.89	M.88
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.87	M.86	M.85	M.84	M.83	M.82	M.81	M.80
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
W21	M.111	M.110	M.109	M.108	M.107	M.106	M.105	M.104
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.103	M.102	M.101	M.100	M.99	M.98	M.97	M.96
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
W22	M.127	M.126	M.125	M.124	M.123	M.122	M.121	M.120
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	M.119	M.118	M.117	M.116	M.115	M.114	M.113	M.112

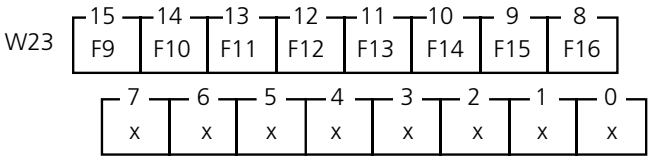
Every message text (0...127, with a max. of 32 characters each) can be allocated to a bit. In addition a specific priority (4 = information, 6 = warning, 8 = fault) and a specific cancel mode (1...4) is assigned to each text. If a message bit is set, to which no message text has been programmed, then it will be ignored.

3.5 EXTENSION AREA

At the moment only extension area W23...W29 is occupied used by the PCS 095, PCS 095.1 (W23...W25). It is reserved for later additions and is therefore used for variables with caution. If however variables are used, then incompatibility with the PLC programme is to be expected.

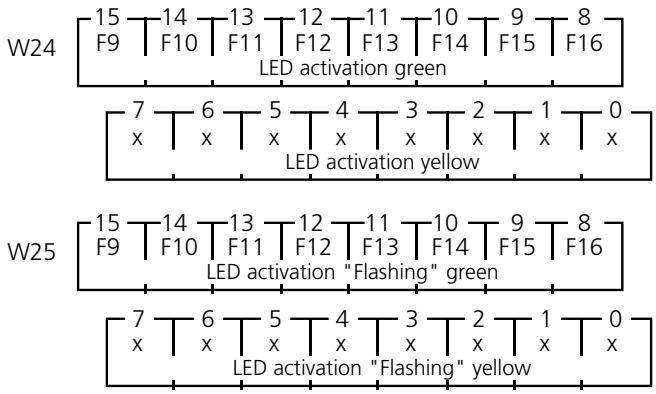
As the PCS 095, PCS 095.1 already uses this area (extra keys W23 and additional LEDs W24 and W25), the PCS 009 or PCS 090 cannot be exchanged against a PCS 095, PCS 095.1, without amending the PLC programme.

3.5.1 EXTRA FUNCTION KEY WORDS (ONLY PCS 095, PCS 095.1)



These key bits are log 1, as long as the corresponding key is pressed and the communication runs without fault.

3.5.2 EXTRA LED STATUS F9...F16 (ONLY PCS 095, PCS 095.1)



For each functions' key "F8...F19", there is a green and a yellow LED available. As there are 2 bits available to every LED, every LED can take on 4 conditions:

- off
- on
- flashing (75% bright phase, 25% dark phase)
- inverse flashing (25% bright phase, 75% dark phase)

If one LED is flashing and another is flashing inverse, then they light up on an exchange basis. A bright phase of one LED corresponds to a dark phase of the other one and reversed.
The state of an LED is set by 2 bits from word W24 and W25 (for the red LED via the function key F11, they are bit 5 of W24 and W25).

Arrangement of the LED states:

W24, bit no. x	W25, bit no. x	Condition
0	0	off
0	1	inverse flashing
1	1	on
1	1	flashing

Example:

W24: 00001111 00001111
W25: 00000000 11111111

-> green LEDs over F9...F12: off
-> green LEDs over F13...F16: on
-> yellow LEDs over F9...F12: inverse flashing
-> yellow LEDs over F13...F16: flashing

3.6 VARIABLES AREA

The variable range is between word W30 and maximum word W255. If this area is not sufficient, then the extension area can also be used (PCS 009, PCS 090: W23...W29 and PCS 095, PCS 095.1: W26...W29). This area is however intended for necessary modifications and therefore is to be used with care (PLC programming!).

If all messages are not required, then the words 15...22 can be used for the variables.

More detailed information about the various variable types can be found in the chapter "VARIABLES/TEXTS/MENUS".

Before the exchange of data, every variable set and actual value must be written in the corresponding words. After the data exchange only the set values must be read back from the respective words.

It is recommended to address variables of the same display page in packets! This also saves PLC cycle time.

The addressing order to the individual variables occurs by programming (PCS*PRO*) of PCS 090, PCS 095, PCS 095.1.

The variable formats STRING, CSTRING, BCD, BIN, WORD and ASCII use the words right aligned and ascending, e.g., BIN-2 on W30...W31 (W30 is HIGH word and W31 is LOW word). The format bit can be used on every individual bit (in order to use every 16 bit in a bit-by-bit manner, 16 bit variables have to be declared). One and the same word can be source and target for several variables even with different format. As a set value, it is recommended to only allot one variable format to a word. One or more actual value variables with differing formats can however at the same time be allocated to this word.

Leading and non used bits in the set values of the types STRING, CSTRING, BCD(0)-1 (length 1...3) and BCD(0)-2 (lengths 1...7) are ignored on reading and on writing back they are set at 0 in the PLC. The type bit only changes the currently addressed bit!

Set values should be occupied as far as possible corresponding to their permitted min-/max-values, before the new start, as they are required as a standard value on which to base the editing. If they are outside the min-/max area, then inverse displays will be presented in the display. As set values, they can first be relinquished after correction procedure.

With a 32 bit variable, the word with the lower number is the higher valued word, and the word with the higher number is the lower valued word.

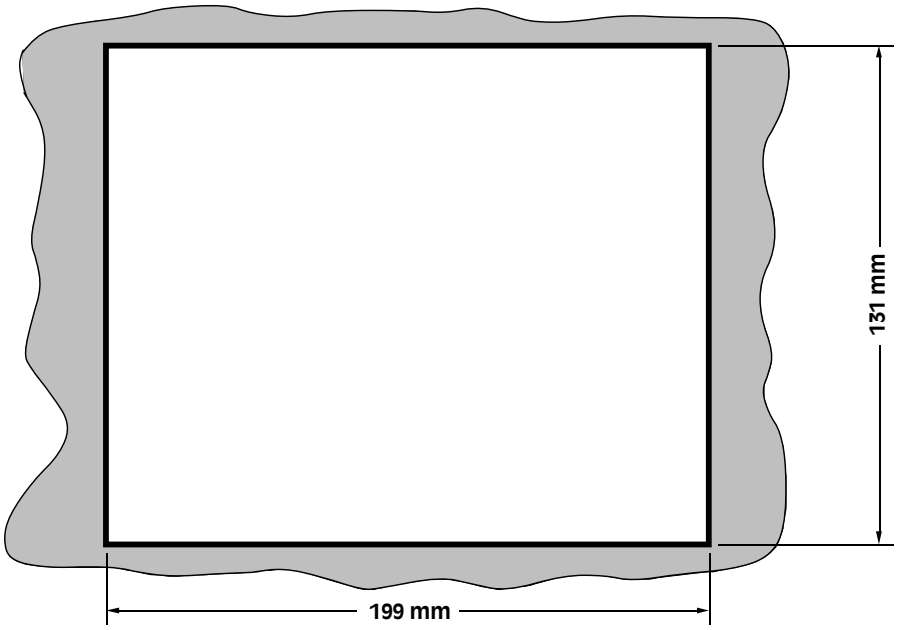
Variables of the type BIT and CSTRING are written into the PLC immediately after their adjustment. All the others are written first after ENT or on leaving the variable range.

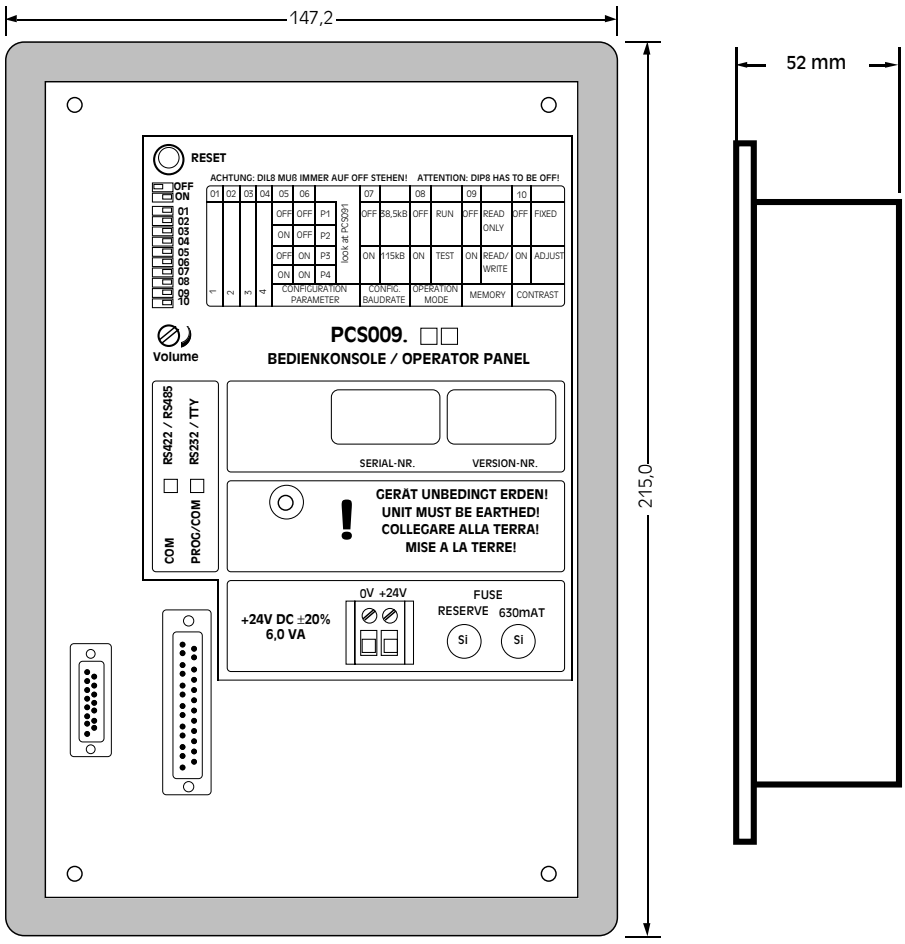
4 TECHNICAL DATA

4.1 TECHNICAL DATA PCS 009

Dimensions:	front panel cut out:	131+1 mm x 199+1 mm
	mouting depth without connector:	52 mm
	external dimensions	147.2 mm x 215 mm
	insertable foil:	98 mm x 13.5 mm
Weight:	1000 g	
Operating voltage:	+24 V DC ± 20%, protected against reversed polarity	
Current consumption:	lav = 250 mA at 24 volt	
Noise immunity:	see manufacturer information	
Protection class:	according to IEC 529, rear: IP 20 front: IP 65 (in a built-in condition)	
Humidity:	0...75% without condensing on the rear	
Vibration resistance:	3 g at 50 Hz in all directions, min 5 hrs 3 g at 100 Hz in all directions, min 1 hr	
Temperature:	storage: -25...+70°C operation: 0...+50°C	
Data storage:	Flash-EEPROM, min 10,000 write cycles	
Front foil:	polyester	
Keys:	mechanical with tactile feedback	
Display:	4 x 20 characters, 5 x 8 matrix, 5 mm character height	
Fuse:	630 mA, small fuse, slow-blow, 1 spare fuse	

4.1.1 DIMENSIONS PCS 009

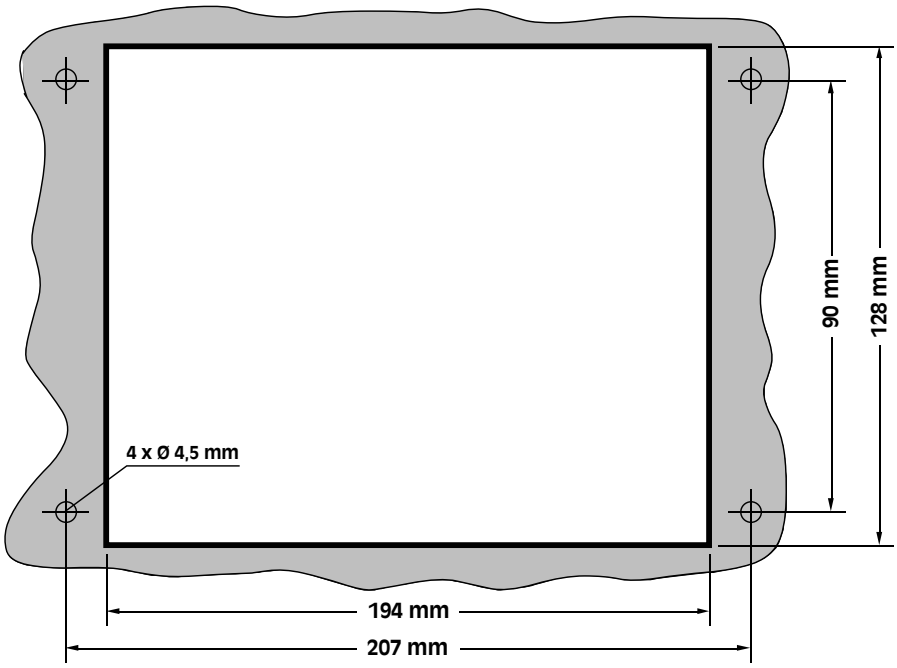


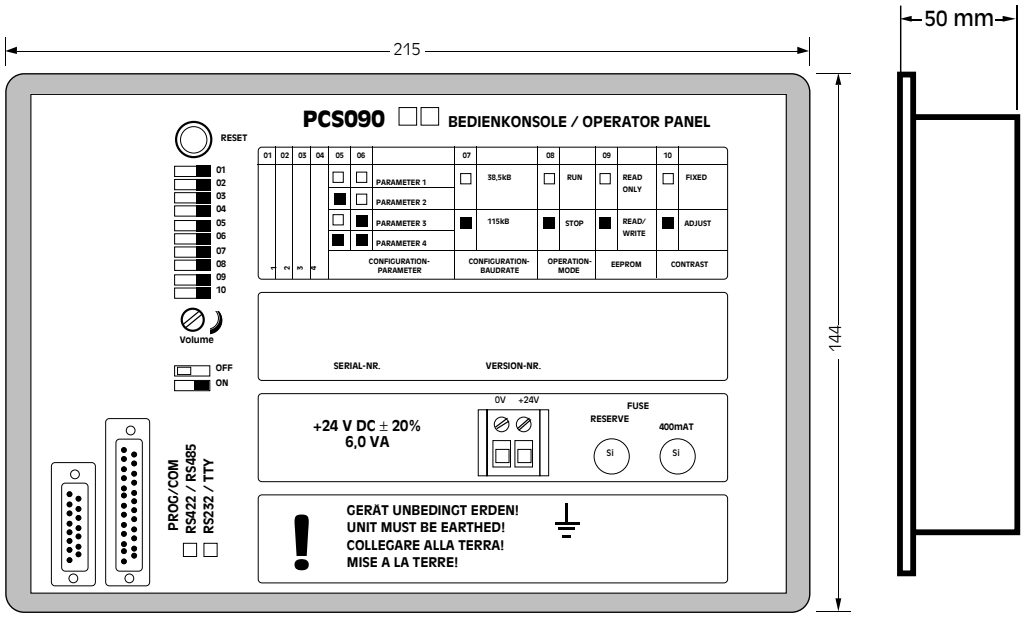


4.2 TECHNICAL DATA PCS 090

Dimensions:	front panel cutout:	194+1 mm x 128+1 mm
	distances of the borings (4.5 mm):	horizontal 207 mm, vertical 90 mm
	mounting depth without connector:	50 mm
	external dimensions:	215 mm x 144 mm
	insertable foil:	186 mm x 14 mm
Weight:	1000 g	
Operating voltage:	+24 VDC ± 20%, protected against polarity reversal	
Current consumption:	Iav = 250 mA at 24V	
Noise immunity:	see manufacturer information	
Protection class:	according to IEC 529, rear: IP 20 front: IP 65 (in a built-in condition)	
Humidity:	0..75% without condensing on the rear	
Vibration resistance:	3g at 50Hz in all directions, min. 5 hrs 3g at 100Hz in all directions, min. 1 hr	
Temperature:	storage: -25..+70°C operation: 0..+50°C	
Data storage:	Flash-EEPROM, min. 10,000 write cycles	
Front foil:	polyester	
Pushbuttons:	mechanical with tactile feedback	
Display:	2 x 40 characters, 5 x 8 matrix, 5 mm character height	
Fuse:	400 mA, small fuse, slow-blow, 1 spare fuse	

4.2.1 DIMENSIONS PCS 090

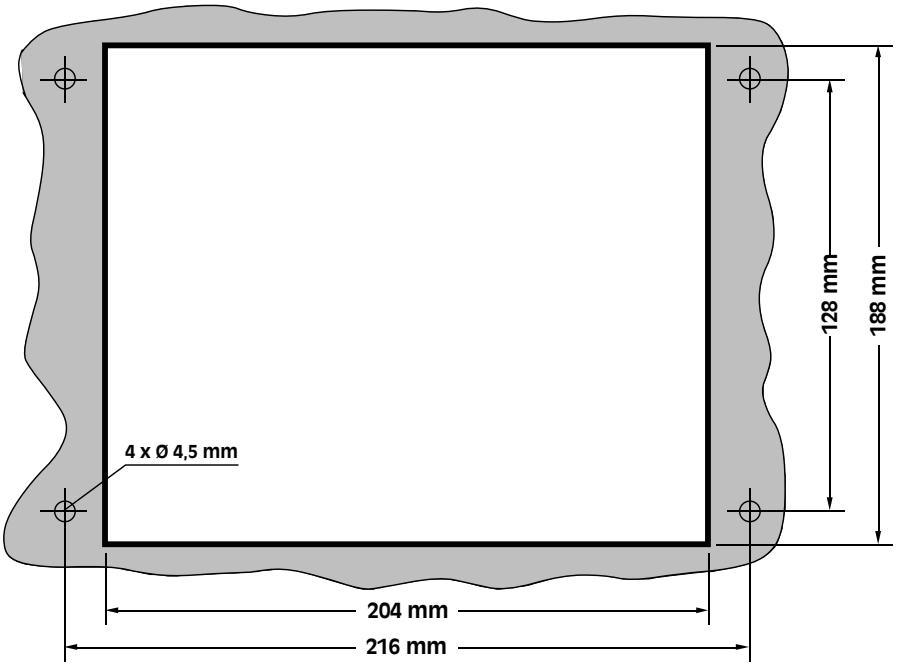


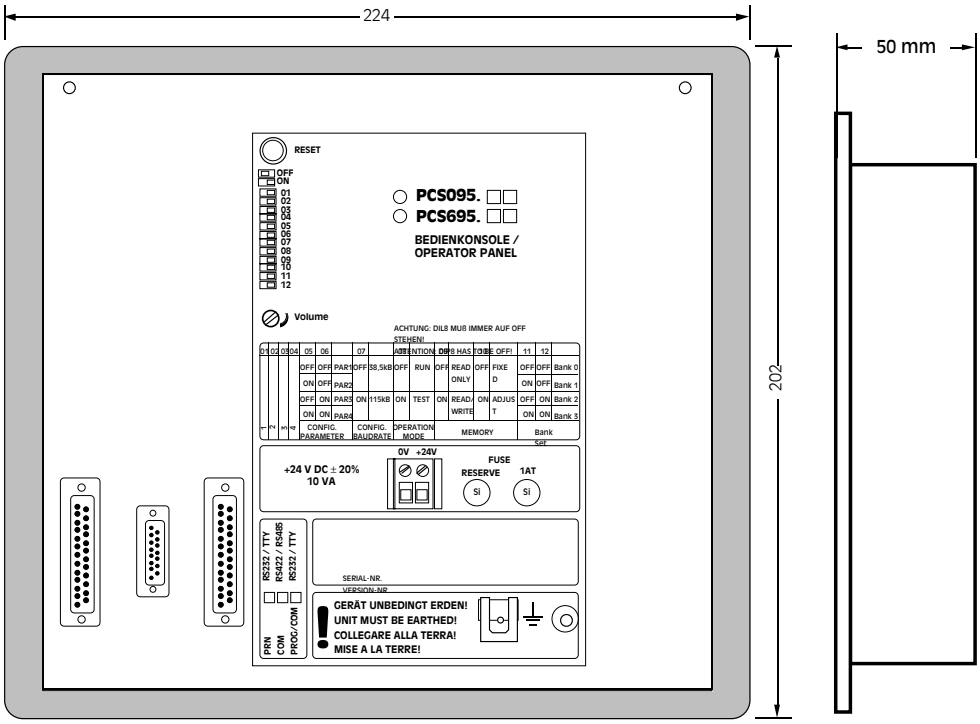


4.3 TECHNICAL DATA PCS 095, PCS 095.1

Dimensions:	front panel cutout:	204+1 mm x 188+1 mm
	distances of the borings (4.5 mm):	horizontal 216 mm, vertical 128 mm
	mounting depth without connector:	50 mm
	external dimensions:	224 mm x 202 mm
	insertable foil:	192 mm x 26 mm
Weight:	1480 g	
Operating voltage:	+24 VDC ± 20%, protected against polarity reversal	
Current consumption:	Iav = 400 mA at 24V	
Noise immunity:	see manufacturer information	
Protection class:	according to IEC 529, rear: IP 20 front: IP 65 (in built-in condition)	
Humidity:	0..75% without condensing on the rear	
Vibration resistance:	3g at 50Hz in all directions, min. 5 hrs 3g at 100Hz in all directions, min. 1 hr	
Temperature:	storage: -25..+70°C operation: 0..+50°C	
Data storage:	Flash-EEPROM, min. 10,000 write cycles	
Front foil:	polyester	
Pushbuttons:	mechanical with tactile tfeedback	
Display:	4 x 40 characters, 5 x 8 matrix, 5 mm character height	
Fuse:	1 AT, small fuse, slow-blow, 1 spare fuse	

4.3.1 DIMENSIONS PCS 095, PCS 095.1





4.4 MEMORY ORGANIZATION

A Flash-EEPROM (electrical erasable static memory) with 32 KB is available in the PCS 009, PCS 090, PCS 095, PCS 095.1 for storing the program (data record), the driver, and other possible functions. Initially a detailed memory map of the required storage space is not available since the occupation depends on the driver (different memory requirements), the possibly used functions, and the number of used variables, texts, and menus. An exceeding of the available memory is already noticed by the comfortable programming software PCS*PRO*. Therefore, you can react well in time.

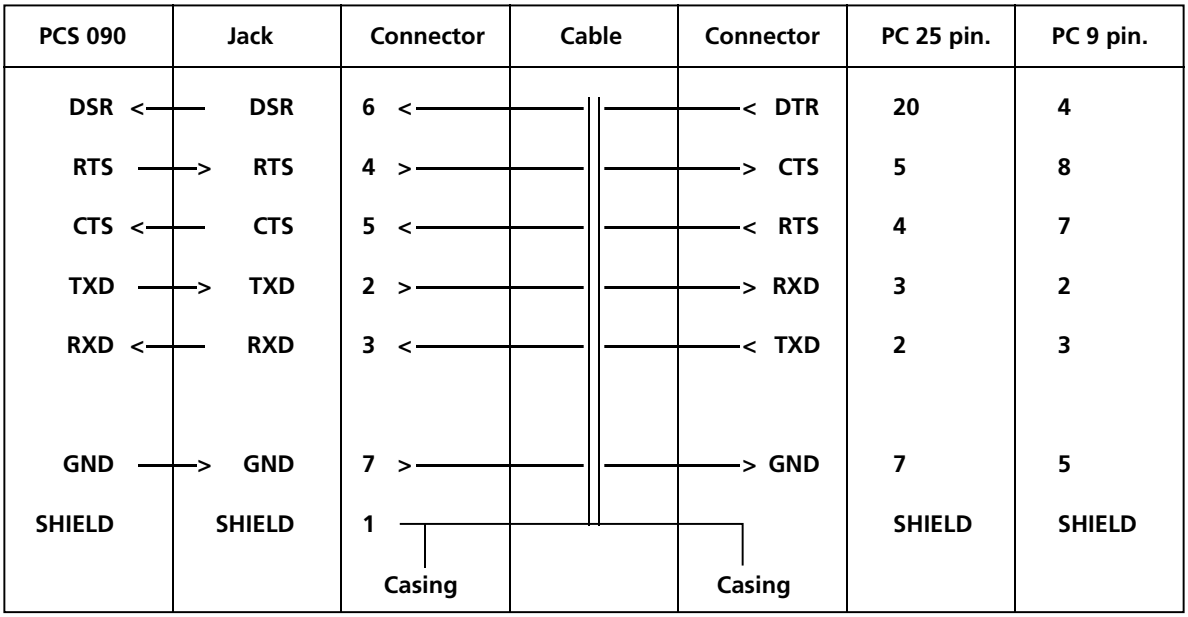
The following applies for the setting up of texts, menus and variables:

- The texts for communication errors are partly in the EPROM programme and in the driver. They cannot be amended. They are set up in the version G in the English language note from.
- The total amount of lines cannot be given, as the texts are filed in a compressed form and the memory for texts, menu nodes and variable inscriptions are dynamic.
- Similar variable inscriptions (character strings) as well as different variables are simply filed. If storage space is limited, then by using similar inscriptions instead of texts space will be saved.
- Every variable occupies 7 bytes in the head, in the trailer, numerous variables require between 6 and 22 bytes. BIT-, STRING- and CSTRING- need 2 bytes per inscription.
- 3 further bytes are necessary per text line (in addition to those for the text itself).
- Every created menu node occupies 8 bytes.

4.5 PROGRAMMING CABLE PCS 733

You require the following cable for the transfer of the program, i.e., of the data record (driver, functions, variables, texts and menus). In addition this cable can be used for the simulation of the PLC to the PC.

Connection PC/PG - PCS 009, PCS 090, PCS 095, PCS 095.1:



4.6 MAINTENANCE

The PCS 009, PCS 090, PCS 095, PCS 095.1 do not require any regular maintenance.

Warning!
Static charge of the front panel is possible. Clean only with a moist cloth.

This is especially important, when using the PCS 009, PCS 090, PCS 095, and PCS 095.1 in an Ex area.

Warning!
The LCD display contains poisonous substances. Do not touch the display, if it is damaged.

4.7 USING THE PCS 009, PCS 090, PCS 095, PCS 095.1 IN AN EX AREA

Warning!
Die Geräte können für den Einsatz in Ex-Zone 1 und 2 von Systeme Lauer nur vorbereitet werden. Je nach Einsatzgebiet ist der Einbau des Geräts nach VDE 0165 bzw. VDE 170/1 durchzuführen.

The PCS 009, PCS 090, PCS 095, PCS 095.1 can be pre-setup for use in Ex area 1 or 2. Depending on the application, the device must be installed according to VDE 0165 or VDE 170/171. For installation in an encapsulated enclosure with pressure protection - including test certifications which may be required - Systeme Lauer offers cooperating companies on request.

This must be specified when ordering the device. A subsequent release or certified declaration by the manufacturer is not possible. The devices can be pre-setup for use in Ex area 1 or 2.

An overpressure encapsulation with a low-pressure system is available. This means that a difference in atmospheric pressure of 2-4mbar exists between the interior space and the outside of the front. Higher pressures may cause damages to the display.

For use of the devices in Ex area 2, please refer to the specifications of the manufacturer and an explanatory memorandum published by Systeme Lauer. The specifications of the manufacturer may be used as basic documentation for the certification of the device in Ex area 2.

5 APPENDIX FOR THE PCS 095.1

The operating console PCS 095.1 features all functions of the PCS 095 and in addition, 4 data records (banks) which can be switched and a **printer interface** for comfortable printer functions.

5.1 OFFLINE MENU

Switching of the 4 data records in the PCS 095.1 is made via an Offline menu.

You access the menu by pressing the HELP and CLR keys. The PCS prompts with the current data record (old bank). Using the + and – keys you can select a new data record (new bank) between 0 and 3. By conforming the selection with ENT the PCS performs a RESET and starts the new data record. By pressing the point (.) key you can exit the menu without performing any changes.

Please note that the data record selection is not kept after a supply voltage cut. After a power-down and power-up of the PCS or after pressing the hardware RESET button (on the rear of the PCS 095.1), the PCS reselects the data record which is specified by the DIL switches 11 and 12 (rear side).

You can generally disable the Offline menu by setting bit 15 of data word 12 to logical 1 during active communication. Under all other conditions, the menu is accessible as described before.

5.2 PRINTER

The PCS 095.1 features a printer interface and supports a printer protocol. The printer parameters are selected using the setup software PCS*PRO* (version 2.1 and up).

5.2.1 PRINTER PARAMETERS

You define the printer parameters using the PCS*PRO* setup software. Some of these parameters are internal variables which are modifiable via a set value menu. An automatic initialization of the printer interface is performed if the PCS 095.1 encounters a change.

The following printer parameters (Zx) are internal variables:

- ZA: printer timeout 1 .. 999 seconds. After this time, the programmable controller is informed about a non-operational printer. This is specially important for printers, e.g. the LCA 710 since this printer sets the RTS line to 0 during printing of a line. Specify a timeout time of 5 seconds for the LCA 750 printer.
- ZB Selection of the RS232/TTY interface.
- ZC handshake selection: NONE, XON/XOFF, RTS/CTS. With handshake set to NONE, the characters to be printed are sent without confirmation - the printer always signals „ready". With XON/XOFF, sending of the characters is only interrupted after receiving XOFF. With RTS/CTS handshaking, sending is only performed if the RTS line is set to 1 level.
- ZD: printer output direction „UP" or „DOWN". Depending on the physical orientation of the printer you can select if the first or last line is transmitted to the printer first. For the LCA printers it is preferable to send the last line first. Thus „UP" has to be selected.
- ZE: baudrate of the printer - selectable between 1200 and 19200 baud.
- ZF: number of data bits of the printer - selectable are 7 or 8 data bits.
- ZG: number of stop bits of the printer - selectable are 1 or 2 stop bits.
- ZH: parity of the printer - selectable are NONE, EVEN, and ODD parity.

In addition, the following functions can be selected in the PCS*PRO* setup software:

- Printer initialization. After a reset, communications loss to the programmable controller, or after a modification of the printer parameters, 0 to 8 characters can be sent to the printer. The default setting is „0D, 0A“ - a line feed.
- Replacement characters for the freely definable display characters. For the PCS you can freely define 8 characters for display but these characters cannot be printed. Therefore 8 replacement characters are to be defined.

5.2.2 PRINTER STATUS

The programmable controller is informed about the status of the printer using bit 15 of DW7. Only with RTS/CTS handshaking, the „Ready“ printer state is clearly defined. With XON/XOFF handshaking, „Busy“ will only be signaled after receiving XOFF. Without handshaking „Ready“ will always be signaled
A communications loss with the programmable controller will always result in aborting the printing.

5.2.3 HARDCOPY

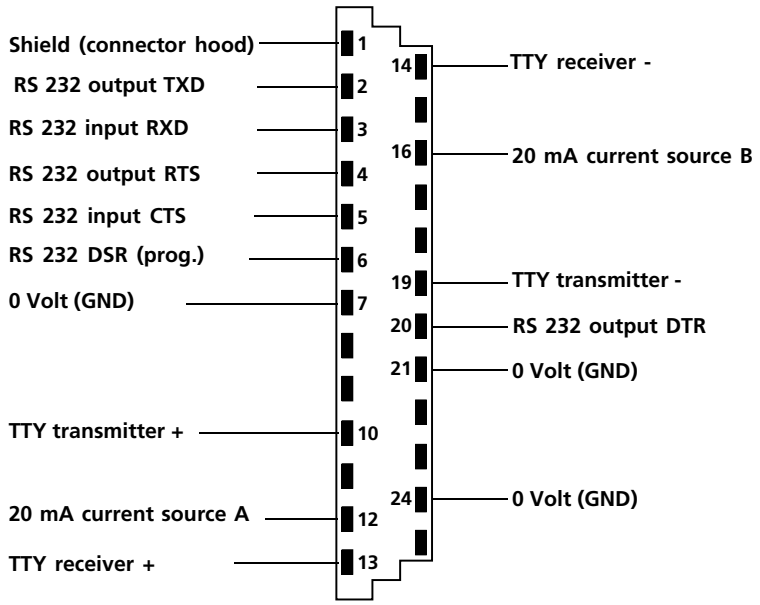
From the programmable controller you can initiate a printout of the current display contents via a hardcopy function. For this, the 0 -> 1 transition of bit 12 of DW 13 is monitored. The contents of the display is copied to the printer buffer and printing is started if the rising edge is detected. Bit 12 of DW 7 is set if the printout has been completed. This bit (12 of DW7) stays set until the requesting bit 12 of DW 13 is reset to 0.

5.2.4 PRINTING OF MESSAGES

From the programmable controller you can initiate a printout of all marked messages. The message word area DW 15 .. 22 is checked for 0 -> 1 transitions if bit 13 of DW 13 is set to 1. The corresponding message(s) are marked for printing if one or more transitions are detected. A currently active printout is signaled via bit 13 (set to 1) of DW 7. All markings are deleted if bit 13 of DW 13 is set to 0. All lines of the messages are printed. The printing process as well as fetching the variables from the programmable controller is line orientated, i.e. the variables are inserted at the time the corresponding line will be printed.

5.2.5 CONNECTOR ASSIGNMENT RS 232/TTY

Connector assignment (rear view on the female connector)



*) currently not used.

Warning!
If external current loop sources are used, the maximum e.m.f. may not exceed 15V. Furthermore, real current sources with a maximum of 22 mA are required. Otherwise malfunctions may occur in the PCS and in the programmable controller!

6 INDEX

- Operating instructions, manuals and software are copyrighted. All rights are reserved. Copying, duplicating and translating in whole or in part is not forbidden. Making of one backup copy of the software presents an exception to this.
- This manual describes our products PCS 009, PCS 090, PCS 095, PCS 095.1. We reserve the right to make changes to the manual at any time without prior notice. The contents of this manual covers the delivery versions PG009.201.0, PG090.202.7, PG095.204.6, PG095.406.8, and PG095.506.8.
- We cannot guarantee the accuracy of programs and data stored on floppy disk nor their error-free condition. We only guarantee that programs are executable within the application described in the manual.
- Since floppy disks are data carriers which can be manipulated we can only guarantee that they are physically undamaged. The liability is limited to supplying areplacement.
- We always gladly welcome any ideas on improvements as well as notes on faults.
- This agreement covers also the special driver manuals PCS 91.xxx. These manuals describe the loadable drivers, the programmable controller handling software and the connection of the PCS 009, PCS 090, PCS 095, PCS 095.1 to the programmable controller.