MAC Operator Terminal

Human-Machine-Interface

User's Manual

MAC E900

Art. No.: 103519
990302
Version A

MITSUBISHI ELECTRIC EUROPE B.V.
FACTORY AUTOMATION

Manual
Operator Terminal MAC E900
Article-No.: 102995

Version			Changes / Additions / Corrections
A	03/1999	pdp	—

About this Manual

The texts, illustrations, diagrams, and examples in this manual are intended exclusively as support material for the installation, handling, and operation of the operating terminal MAC E900 in combination with a MELSEC, Allen Bradley or SIMATIC S5/S7 programmable logic controller. Optionally the operator terminal can be operated via an interface card within a PROFIBUS DP network.

If you have any questions concerning the installation and operation of the equipment described in this manual, please contact your sales office or one of your sales partners (see backpage of cover).
You can find current information and answers to often received questions on our fax-back system MEL-FAX (Fax-No.: Germany (0 21 02) 486 485 or (0 21 02) 486 790).

Safety Information

For use by trained personnel only

This manual is only intended to be used by trained and fully-qualified electrical personnel who are familiar with the safety standards for automation systems. Project planning, installation, commissioning, maintenance, and testing of the equipment may only be performed by trained and fully-qualified electrical staff who are familiar with the safety standards for electrical devices and automation systems.

Proper use of equipment

The operator terminal MAC E900 is only intended for the uses specified in this manual. Strictly observe all the instructions and specifications provided in this manual. Only accessories and peripherals explicitly approved by MITSUBISHI ELECTRIC may be used in combination with the following programmable logic controllers: MELSEC FX family and MELSEC A and Q series.

Any other usage of the equipment not confirming to this definition shall be deemed to be improper use.

Relevant safety regulations

All the relevant safety and accident prevention regulations must be observed in all stages of project planning, installation, commissioning, maintenance, and testing of this equipment.

Observance of the following regulations is particularly important (this listing does not necessarily cover all regulations relevant for you):

- VDE regulation
  - VDE 0100
    Regulations for the installation of high-power electrical systems with rated voltages up to 1.000V
  - VDE 0105
    Operation of power systems
  - VDE 0113
    Safety of machines; electrical equipment of machines
  - VDE 0160
    Electronic equipment for high-power electrical systems
  - VDE 0550/0551
    Regulations for transformers
  - VDE 0700
    Safety of electric devices for domestic or similar uses
  - VDE 0860
    Safety regulations for mains supplied electronic devices and their accessories for domestic or similar uses
- File prevention regulations
- Accident prevention regulations
  - VBG Nr. 4
    Electrical systems and equipment

Danger symbols

In this manual, warnings of hazards and potential problems are identified with the following two symbols:

DANGER:

This symbol indicates that failure to observe the safety procedures described can result in severe electrical shock hazards that can cause serious injury to the operating staff. In some case, such as hazard can also be lethal.

CAUTION:

Failure to observe the procedures identified by this symbol can result in incorrect settings or damage to the equipment or other property.

General safety information

DANGER

- ***After delivery inspect the device for damages in transit. In case you detect any damage, please inform your supplier.***
- ***This product complies with the requirements of article 4 of the EMC regulation 89/336/EEC.***
- ***Never use this device in an explosion hazardous environment.***
- ***Any changes and modifications of this device are prohibited.***
- ***Exclusively use MITSUBISHI ELECTRIC approved spare parts.***
- ***Read this manual carefully prior to installation.***
- ***This device must only be operated by qualified personnel.***

Safety information on installation

DANGER

- ***This device is constructed for operation on a fixed location.***
- ***This device has to be installed following the supplied manuals.***
- ***This device has to be earthed following the supplied manuals.***
- ***This device has to be installed by qualified personnel.***
- ***High voltage, signal, and supplying lines have to be wired separately.***
- ***Never expose to direct sunlight.***

Safety information on operation

DANGER

- ***Always keep this device clean.***
- ***Never operate the EMERGENCY STOP and other safety functions from the MAC operating device.***
- ***Make sure that key buttons, screen, etc. do not get in touch with sharp-edged items.***

Service and maintenance

Only the enclosed warranty conditions are applicable.

CAUTION

- ***Clean the display and the front panel with soft cloth and a mild cleansing agent.***
- ***Only use the batteries specified by MITSUBISHI ELECTRIC. Only qualified personnel may replace the batteries. The person replacing the batteries has to be earthed, e.g. via an earthing strap upon the wrist.***
- ***Only qualified personnel may repair this device.***

Dismounting and disposal

CAUTION

- ***Regarding a – even partial – recycling of this device the local regulations are applicable.***
- ***Please remind that the electrolyte capacitor and the display unit contain harmful substances.***

Contents

1 Introduction

The pace of production in today's industrial work environment increases steadily. At the same time, the operator's job responsibilities increase and become more comprehensive. Quickly and simply, the operator must get machine status information and, equally simply, must be able to control the machine's operation. The PLC system's functions increase and become more advanced, making it possible to effectively and efficiently control complicated processes. With the *operator terminal* from Mitsubishi Electric Europe B.V., *human-machine* communication is made simple and safe – even for the most advanced processes.

The graphical operator panels were developed to meet the requirements of man-machine communication for supervision and control in a variety of usage areas in the manufacturing- and process industries, etc. The terminal simplifies the operator's work because it is easily adapted to the operator's work environment. This means that the operator can continue to use the terms and protocols he or she is familiar with.

In the terminal, *projects* can be built up as *menu hierarchies*, or as *sequences*. A menu hierarchy comprises a main screen (with, for example, an overview) and a number of subsequent screens with more detailed information for a particular section. The operator can select which screen is displayed.

In the operator terminal, a screen is called a *block*.

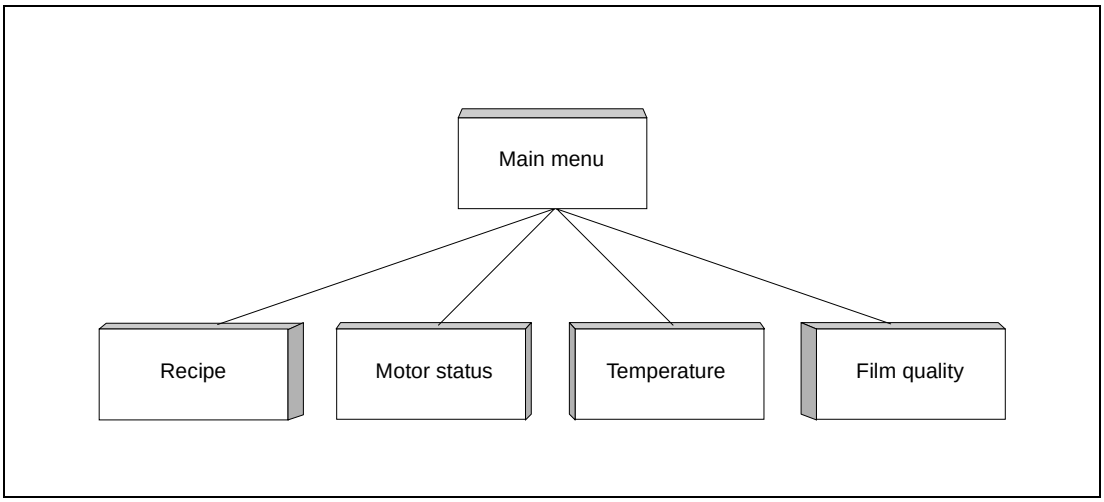


Fig. 1-1: *Menu hierarchy*

A sequence starts with a main menu, from which the operator chooses a sequence where the blocks are displayed in a pre-set arrangement. Normally, the blocks are controlled by the PLC program.

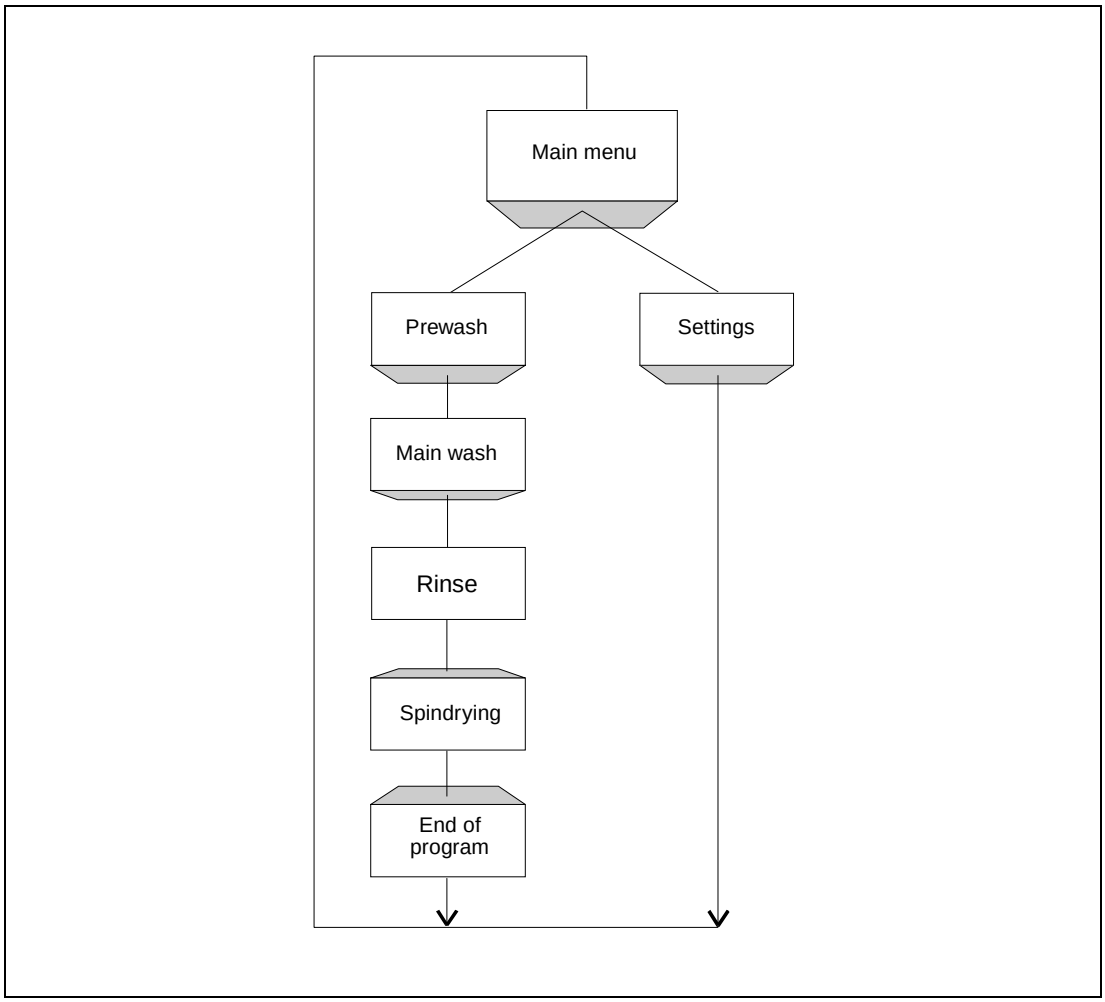


Fig. 1-2: *Sequence*

The terminal's functions enable either graphical or alphanumeric *presentation* and *control*. Additionally, there are functions for *alarm handling, report printout, realtime trend, recipe handling, time control*, and more.

The terminal's functions are not only simple to manage, they also provide clear cost advantages compared to conventional systems with buttons, indicator lamps, time relays, preset counters, and seven-day clocks. The terminal also offers functions which facilitate a more comprehensive use of the PLC system.

1.1 Programming

The terminal is programmed through a PC using the *MAC Programmer+/SW-MTA-WIN* software. Hereafter this product is referred to as the *PC software*. It is also possible to program the terminal through the built-in keyboard.

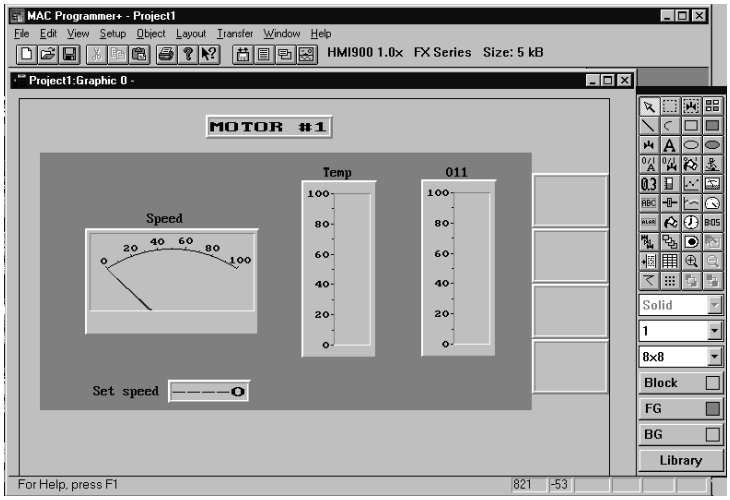


Fig. 1-3: *Programming with a PC*

For the most part the terminal is *object oriented*, that means its work is based on an object, and the functions this is to have are determined afterwards. All types of signals are defined on this principle.

The programmed project is stored in the terminal.

1.2 Connecting the terminal to the PLC system

A terminal couple to the PLC system offers many benefits:

- the user doesn't need to make changes or additions to existing PLC programs
- the terminal doesn't block any of the PLC system's inputs or outputs
- it improves the clarity and accessibility of such PLC system functions as time control and alarm handling.

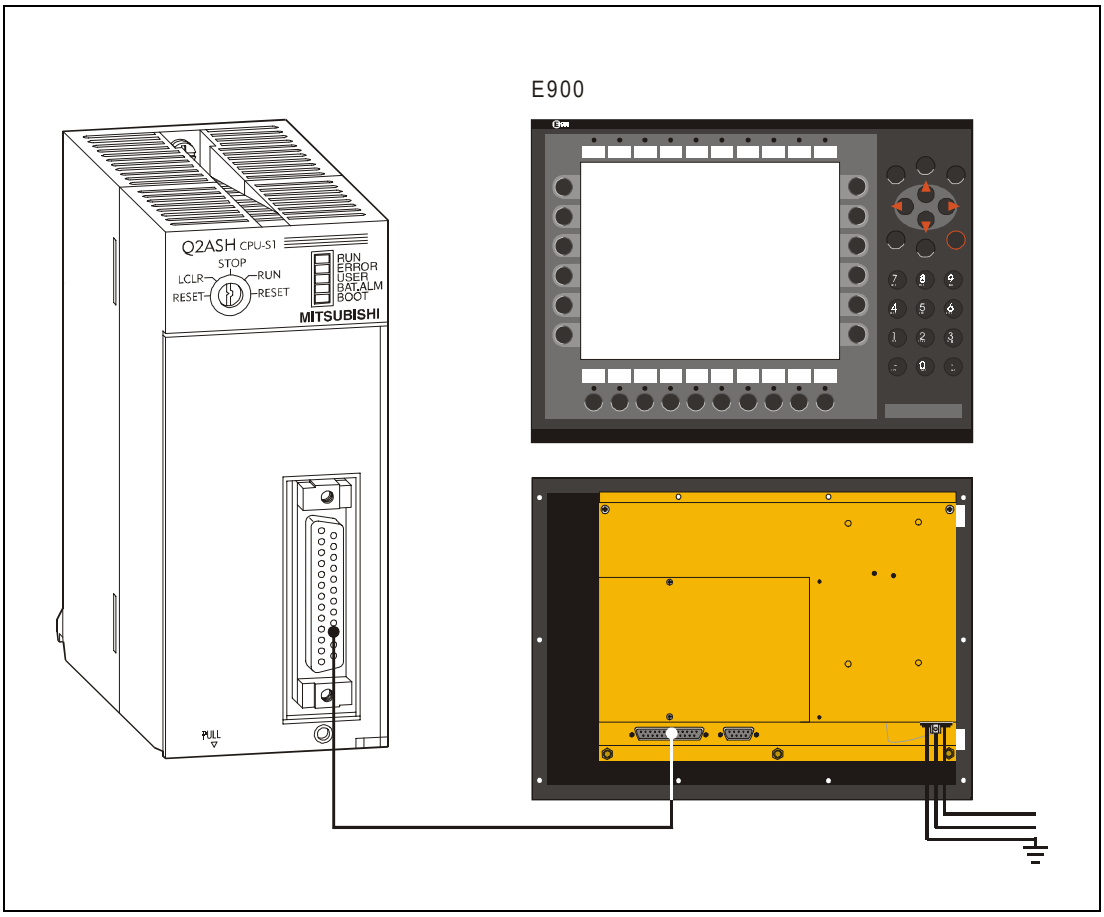


Fig. 1-4: *Connecting the terminal to a PLC system*

1.3 Operation modes

E900 has three operating modes:

- *Setup mode,* where the basic settings are made such as choice of PLC system, menu language and printer settings.
- *Programming mode*, where the terminal can be programmed and changes can be easily made at system set-up.
- *Run-time mode,* when the terminal is running with the PLC.

Switch between PROG and RUN

Switch between programming, PROG, and run-time, RUN, by simultaneously pressing the keys [←] and [MAIN]. The possibility to switch between PROG and RUN from the terminal can be controlled via a password. For more information we refer to chapter 4, *Programming using the PC software*.

Switch between PROG and Setup

To switch from PROG to Setup you press [LEAVE] until the start-up menu is displayed, then press any key. Press [LEAVE] to return to PROG.

Switch between RUN and Setup

Press [←] and [MAIN] simultaneously to enter programming mode, PROG. Then press any key when the startup menu is displayed to enter the Setup mode. To switch from Setup mode to RUN you press [←] and [MAIN].

1.4 Status display and control

The operator is very familiar with indicator lamps, as well as analog and digital displays for showing status as they appear today in the vast majority of applications. The same applies to push buttons, rotary and thumbwheel switches for controlling a system. Replacing them with a terminal provides the operator with all status display and control systems in one unit.

The operator can see and control all the information in the PLC system using the nomenclature of the machine. By doing so the operation of the complete system is not only enhanced but also simplified.

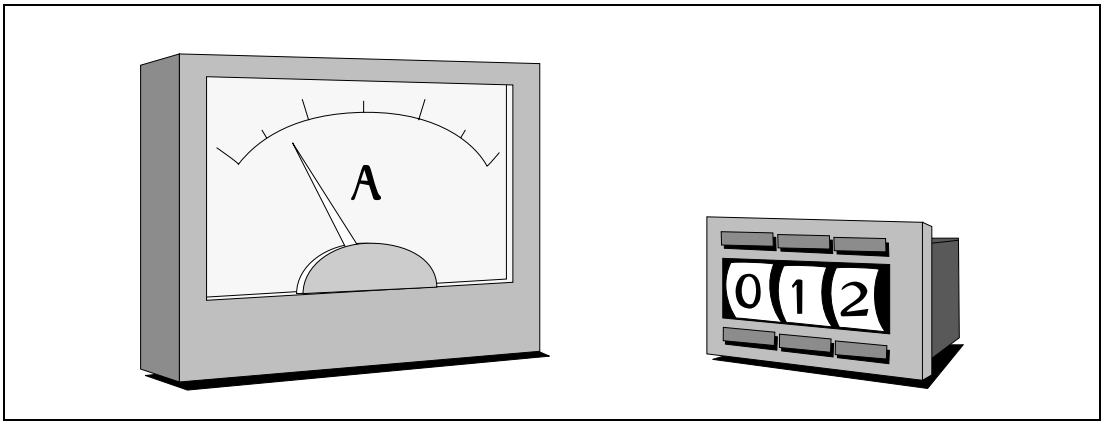


Fig. 1-5: *Analog and digital display*

This is possible with one terminal thanks to the fact that the interchange of all information takes place through so-called blocks in the terminal.
A block can be defined as a textblock with only text information, or a graphic block with full graphics.

The terminals have function keys for direct control. When an operator presses/energizes a function key during the run-mode, the digital signal in the PLC system linked to the function key will be switched on.

If several blocks are used, the operator can use jump objects to move between different blocks. In this way, a menu tree is created, allowing a structured application.

1.5 Location of the terminal

The terminal should be located close to the machine, to have maximum usability. The operator then receives the correct information directly and can react quickly and efficiently.

In some cases, such as monitoring/running or maintenance, the operator's workplace can be several kilometres from the process. Communication takes place via modem.

Several terminals can be installed, if the production line is long with a large number of work stations.

A parallel working unit can also be connected. This could be another terminal. It could also be a PC with the programming tool MELSEC MEDOC. It is then possible to program the PLC system via the terminal at the same time as the terminal communicates with the PLC system.

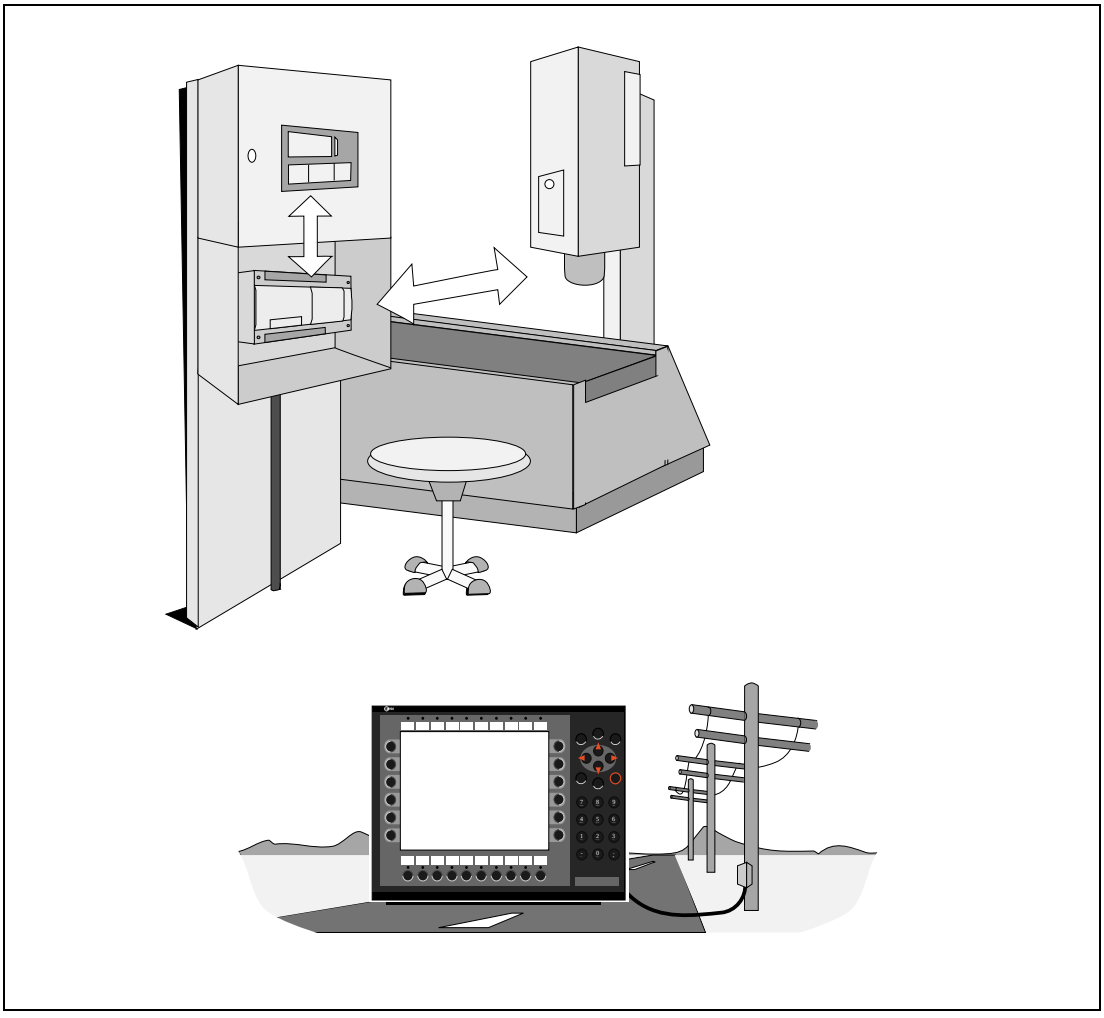


Fig. 1-6: *Location of the terminal*

1.6 Compact solutions

External units such as barcode readers, weighing machines, telephone modems and so on can be connected to the PLC system via the terminal. All the connection requires is that the unit can be connected to a RS232 interface and that the communications is by ASCII-protocol. Data entering the terminal is written directly to the PLC register.

For information about installation and technical data, see the Installation manual delivered with the terminal.

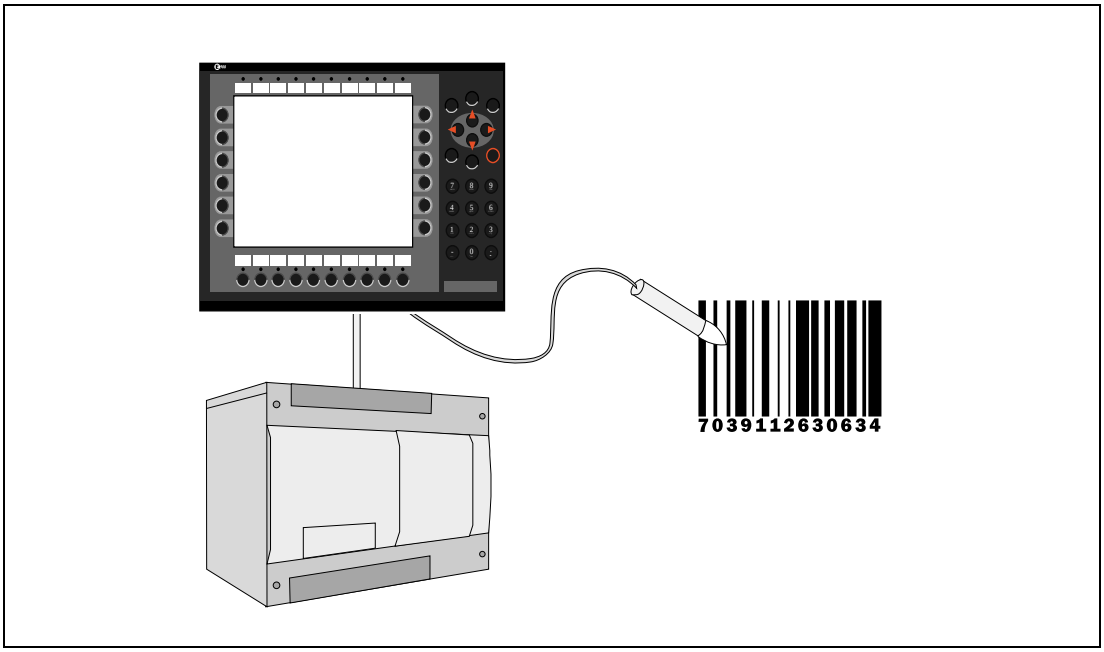


Fig. 1-7: *Connection of external units*

2 Function overview

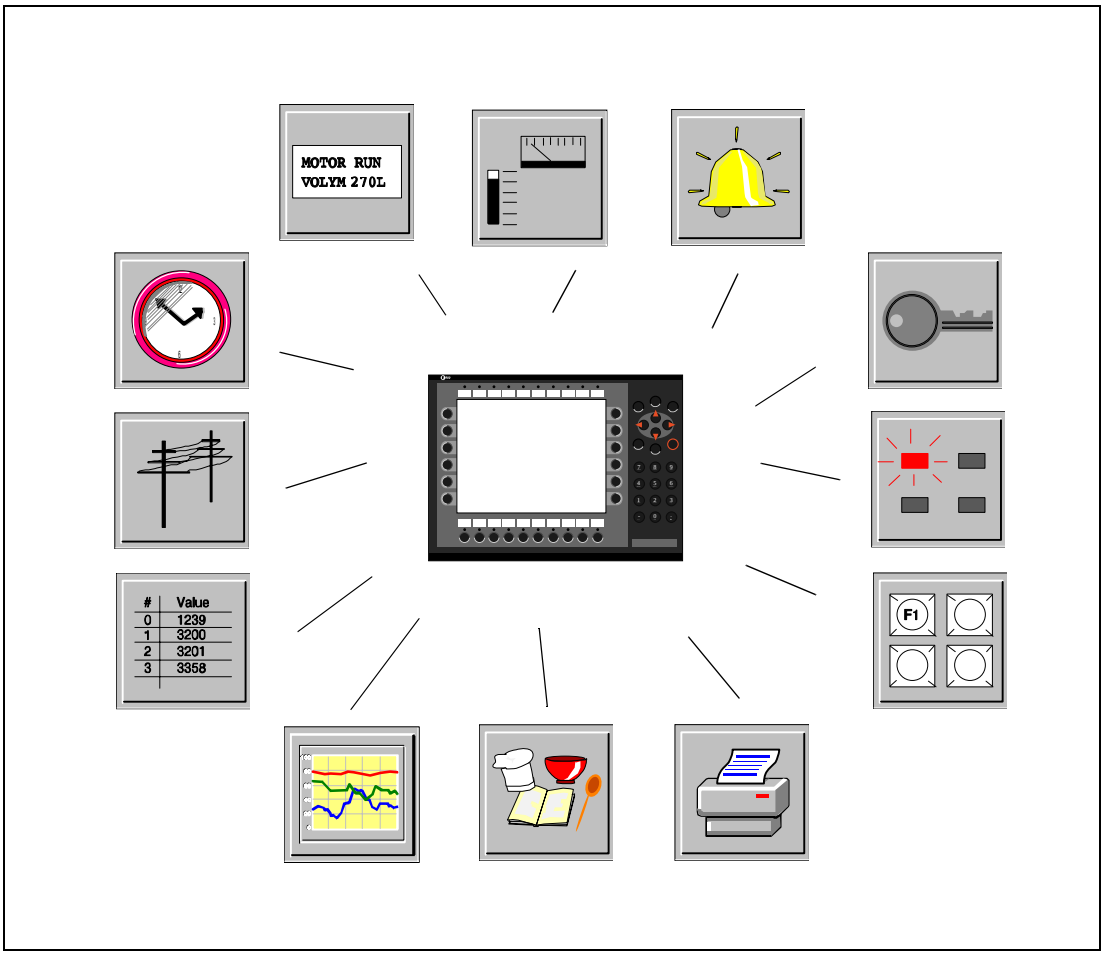


Fig. 2-1: *Function overview*

2.1 Basic functions

2.1.1 Graphic block

Graphic blocks are used to create graphic operator dialogs. There are static and dynamic graphics.

Static graphic objects

Static graphic objects form the background graphics such as lines, circles, symbols and texts in various sizes, e.g. headings and help texts. Static information is not changed by the PLC system or the operator.

Dynamic graphic objects

Dynamic graphics are graphics which are linked to signals in the PLC system. Examples of dynamic graphics are bar graphs, trend curves, analog and digital clocks, VU meters etc. Dynamic information such as numeric values and turning on/off are changed by the PLC system or the operator.

2.1.2 Textblock

A textblock is used to build up dialogs and reports. The textblock can consist of an arbitrary number of text lines.

Static text

Static text is a textblock text which is not changed by the PLC system or the operator when executing the program.

Dynamic object

Dynamic objects are objects which are linked to the signals in the PLC system. There are eight types of dynamic object: digital, analog, jump, date/time, bar graph, multi-choice, message and text object. Information to the object is changed continuously by the operator or PLC system.

2.1.3 Alarm handling

In certain cases events can occur in the process which require immediate attention. To meet this requirement there is the possibility in the terminals of linking the signals to an alarm message. Maintaining alarm management is important in order to minimize downtime costs. During operation, the alarm is registered in an alarm list, with time stamp. With the help of the list the operator can trace the cause and sequence which activated the alarm, and can also acknowledge each alarm. The alarm list can be displayed or printed out. Alarms can be grouped, and can contain dynamic data.

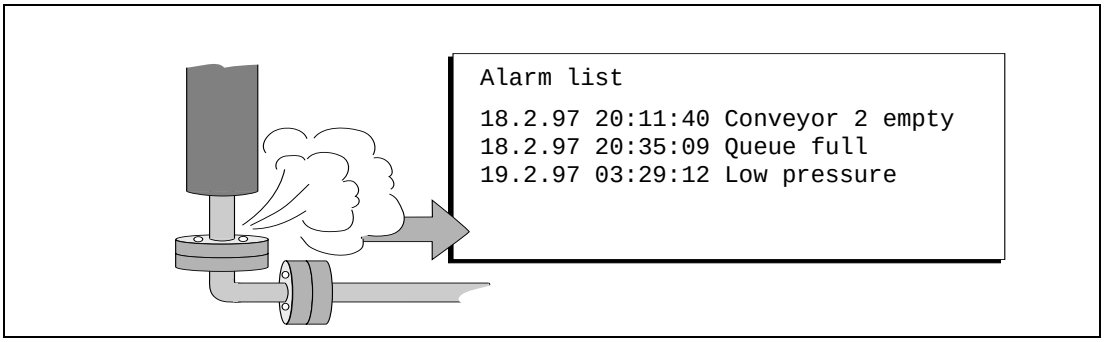


Fig. 2-2: *Alarm message*

2.1.4 Password

Security levels can be assigned to blocks, function keys, and maneuvrable objects, to restrict access to, for example, certain parts of an installation, via a password, the user can log in to the approved level.

2.1.5 LED

Light diodes are used to indicate different events in the process, such as on/off. The terminal has 20 LEDs which can assume three shades: off, red and green. The light diodes are controlled by the data register in the PLC system.

2.1.6 Function keys

Function keys are used to activate different events in the process; e.g. to start a fan. They can be linked to digital signals which are switched on during operation when a function key is pressed. Function keys can be defined for global use - in the whole project - or local, in a particular block. Function keys can also be used for internal functions, for example, to activate change of blocks.

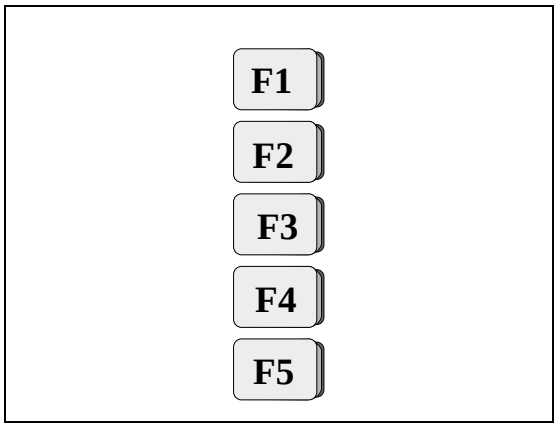


Abb. 2-3:
Function keys

2.1.7 Historic trend

This function permits analog values from the PLC system to be collected and presented in the terminal during operation as a curve diagram. It is possible to present several independent trend curves in the same or different blocks. The number of curves possible is determined by project memory allocation.

2.1.8 Recipe handling

Recipe handling permits fast change of product in the production line. For example, a production line produces several similar products, such as ice cream with different flavours. The same project blocks can be utilized, with changes made to select parameter.

The function rationalizes time-critical production by making it possible to quickly get new parameter settings from the PLC system.

A portion of the project memory can be set aside for recipe storage. This function means that all dynamic data in a block (such as signals, and their values during operation) are saved in a file. With the function, parameter settings can be re-used. The operator creates a recipe library with files containing the different parameter settings.

2.1.9 Report printout

Text blocks can be used to follow up and facilitate a more effective production, by creating regular reports of, for example number of manufactured units, or production stops. Reports can be printed out, or their information can be forwarded to a supervisory computer, for summarization or further processing.

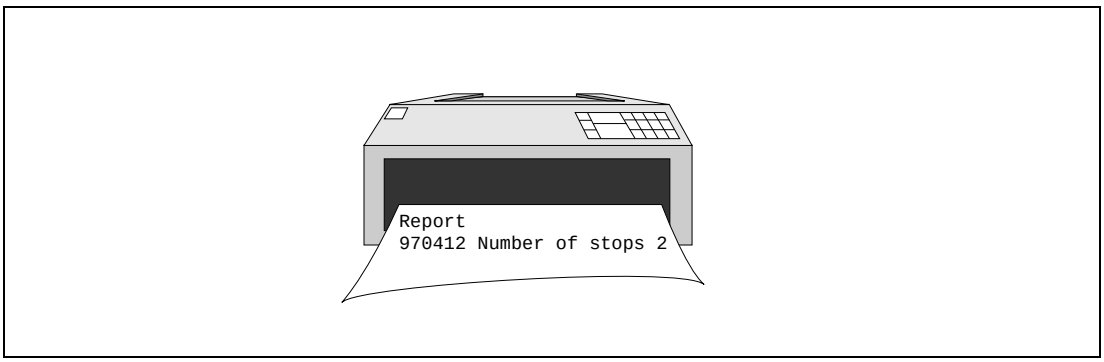


Fig. 2-4: *Report printout*

2.1.10 Communication

The terminal can be used in several communication configurations: field buses, transparent mode, no protocol mode, network with IFC 50E and in PROFIBUS network.

Transparent mode

Transparent mode is used to connect parallel units to the PLC system. In this way, several terminals can work with the same PLC system, or a PC with MELSEC MEDOC or a supervisory PC operating system via the terminal.

Network communication

Network communication via IFC 50E is used to connect more than two terminals to an PLC system, while still retaining good operating performance. An example is production lines with a terminal at each work station. A network can include one master and up to seven slaves. The terminal connected to the PLC system is the master. The separate manual for IFC 50E contains more detailed information.

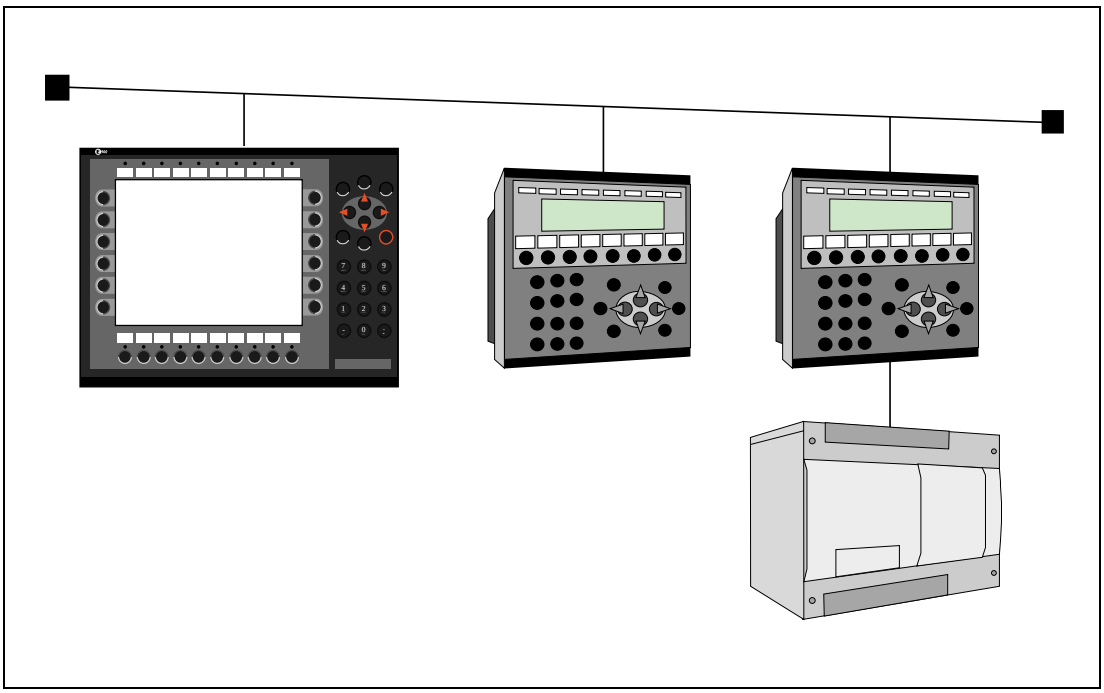


Fig. 2-5: *Network communication*

The IFC PBDP expansion card enables the terminal to be connected to a PROFIBUS network. With PROFIBUS, it is possible to communicate with any PLC system; for example Siemens. The separate manual for IFC PBDP contains more detailed information.

No protocol mode

This mode is used to link different PLC systems, or to connect external units to the PLC system, such as bar code readers or weighing scales. Communication takes place via control block.

2.1.11 Time control

Many applications also require time control. Via the terminal it is possible to control when a machine is on or off, instead of connecting to extra time relays or seven-day clocks.

A real-time clock is built into the terminal. Digital signals are set and reset relative to this clock without requiring programming in the PLC. Control can then be carried out with respect to the time of day, and day of the week.

Four different time intervals can be entered for each object. A time interval is a day of the week or a part of a day as well as the start and stop times. It is possible to allow the operator to change the time interval dynamically.

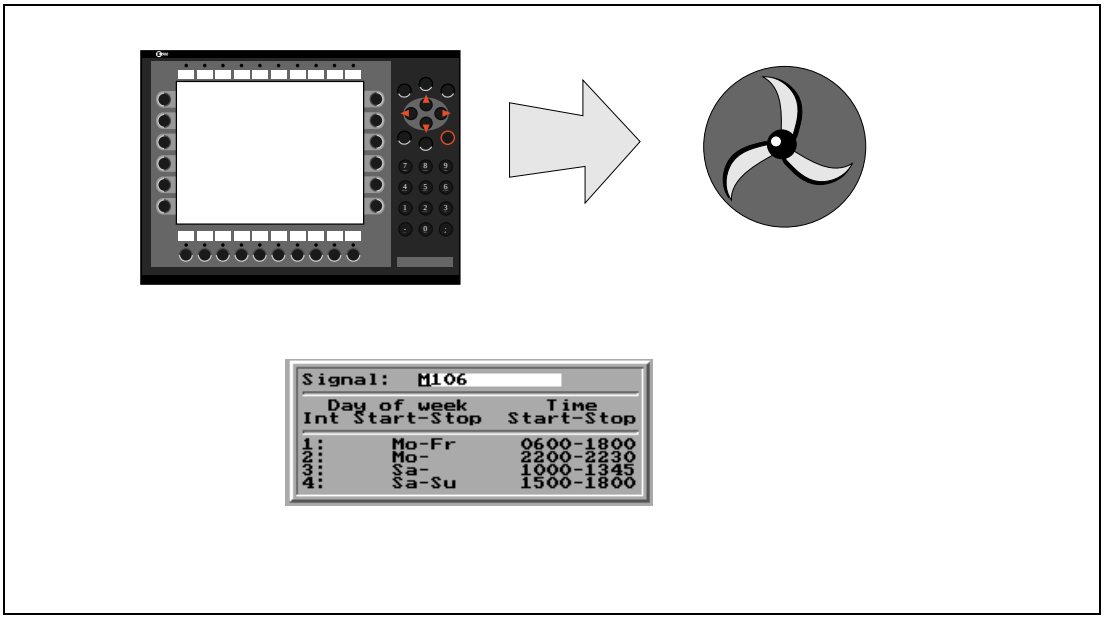


Fig. 2-6: *Time control*

3 General

This section describes how the terminal is designed as well as the basics for using it. Furthermore there is an explanation of the general rules and object parameters as well as common functions which apply to the terminal.

NOTE

In the manual we use a PLC-system from Mitsubishi Electric. For other PLC systems we refer to chapter 20, *Connection to an Allen Bradley PLC system* or chapter 21, *Connection to a Siemens PLC system* and to the manual for the PLC system respectively.

3.1 Method for programming a project

Building up the application in the terminal graphically provides the operator with concise information on the operation of the system. It is important to organize your application well. Think about which functions are required. Begin at the introductory level and then work down through the levels of detail. When an object is programmed base this on the functions in your application. Each function then becomes one or more blocks depending on how complex the function is. A project can contain both graphic- and textblocks and each block can contain static and dynamic objects. To achieve a structured application the block should be created in a hierarchy which makes working natural for the operator.

In the run-time mode it is possible to test all or parts of the application before running.

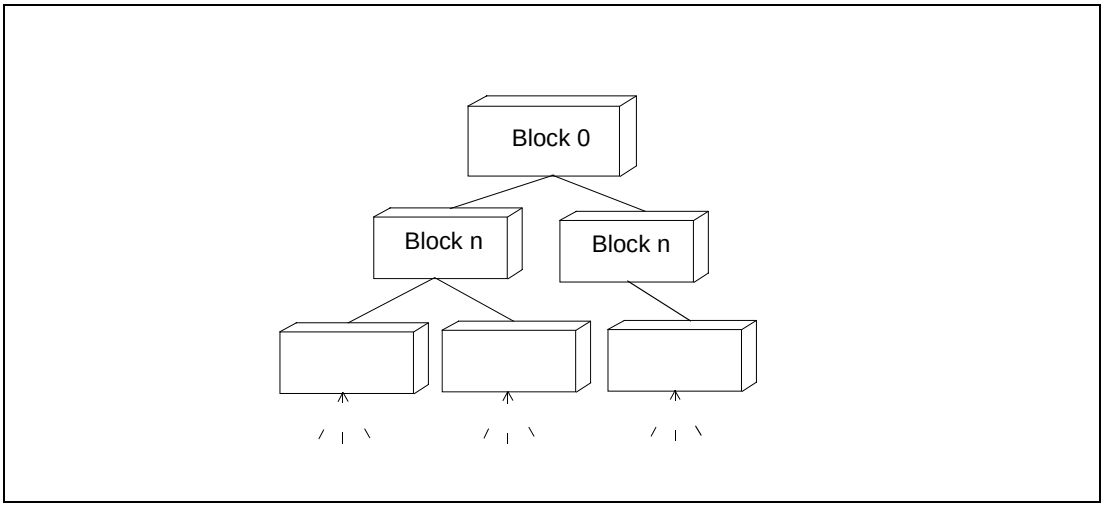


Fig. 3-1: *Structured application*

3.2 General advice

3.2.1 Communication

The following should be noted so that communication is quick and efficient.

Group PLC signals consecutively

The terminal is continually reading signals from the PLC according to the following:

- Display signals
- Block print-out signals
- LED register
- Trend register
- Bargraph register with min/max indicators
- PLC clock register
- Alarm

The signals in the screens are only read when the screen is on, that is the number of blocks does not affect the communication time.

The signals from the PLC system are read most rapidly if the signals are consecutive. If for example, 100 alarms are defined, it is quickest to read these if they are linked to, for example, M0–M99. If the signals are spread out (e.g. X0, Y30, T45 etc.) the updating is slower.

The following signals do not affect communication:

- Signals linked to function keys
- Time channels
- Objects in the alarm messages

Efficient block changes

Block changes are carried out most rapidly and efficiently through the block jump function on the function keys or through a jump object.
”Display signals” in the block header should only be used when the PLC system is to force the presentation of another screen.

The ”New Display” register can also be used if the PLC system is to change the screen. This does not affect communication as much as a larger number of ”Display signals”.

Use the terminal´s clock

An extra load is put on communication if the PLC system´s clock is used since the clock register must be read up to the terminal. Downloading of the terminal´sclock to the PLC system also creates an extra load. The interval between downloadings should therefore be as long as possible.

3.2.2 Operator´s pictures

Use graphic blocks for operator communication

The textblock is designed in the first place for report print-outs. They are slower and demand more memory than graphic blocks.

Use the 3-D effects for elegant operator pictures

Combinations of objects with frames and 3-D rectangles can be very effective. In this context the ”lighting” appears to come from the top left-hand corner, that is the ”shadowing” is on the under and right-hand edges of raised objects and on the upper right-hand edges of inset objects.

3.3 Menu structure

The terminal is divided into three modes: *configuration*, *programming* and *run-time mode*. Normally configuration and programming are carried out with the PC software. Each mode consists of different number of menu levels depending on the function. Each level consists of a menu where you make a choice or state a parameter to go on to the next level (menu).

The application is built up of *blocks*, *graphic blocks* and/or *textblocks* (primarily for report printouts). Values from the PLC system are shown and altered in the blocks. Each block has a block number between 0 and 989 allocated by the program. Blocks 990–999 are reserved for special purposes, so-called *system blocks*. The terminal is object-oriented which means that a block can contain all the signals linked to an object for controlling and monitoring; a pump for example.

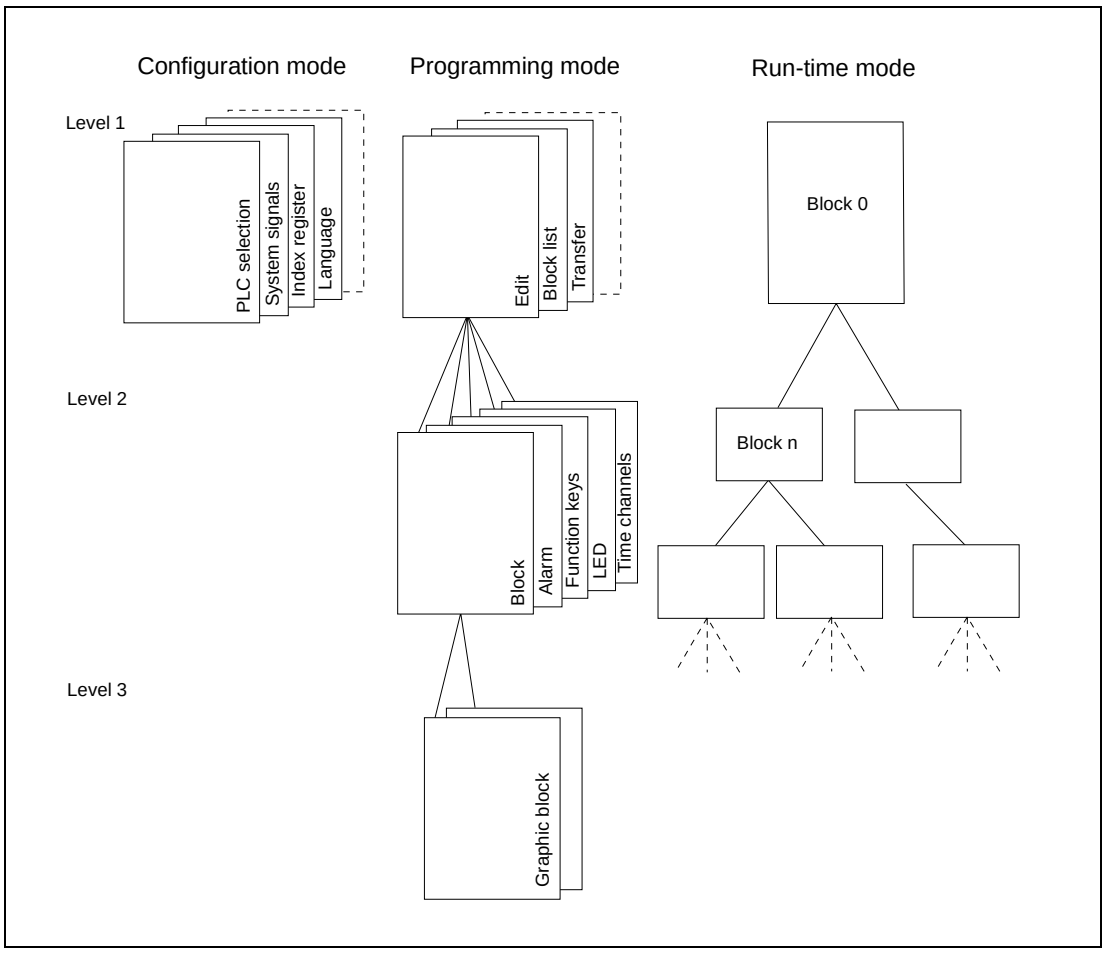


Fig. 3-2: *Menu structure*

3.4 Function keys

The figure below shows how the function keys are numbered.

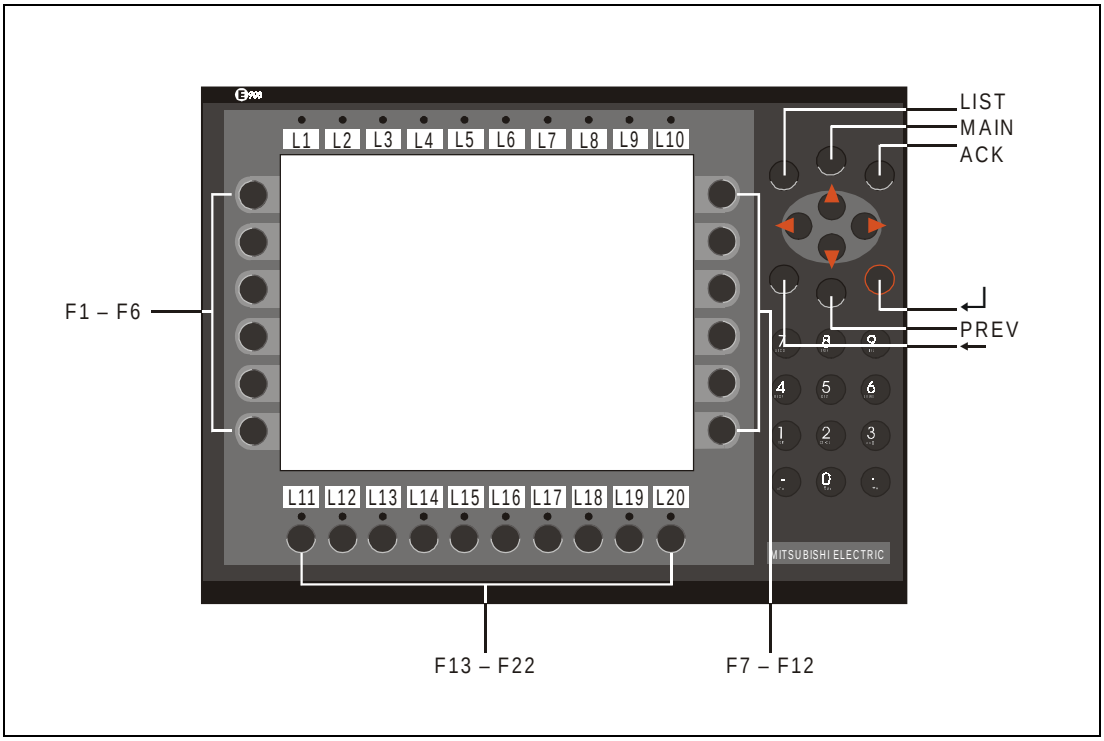


Fig. 3-3: *Function keys*

3.5 Blocks

A block header is defined for each block. The block number, type of block, status word etc. are contained in the block header. The alarm, time channels and contrast settings functions can also be called up as blocks. These are called system blocks.

NOTE The block type cannot be altered for a defined block.

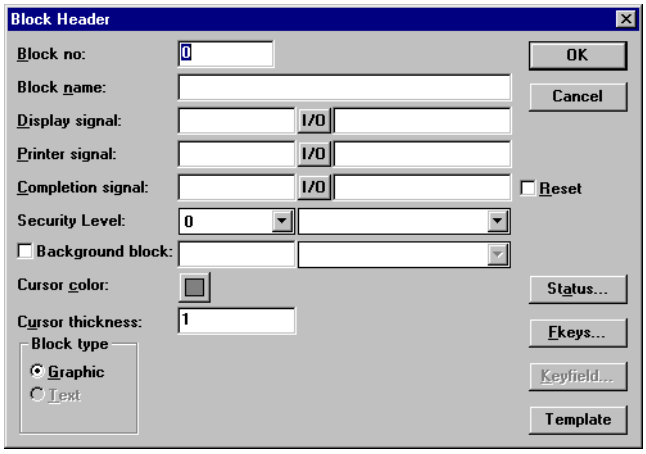


Fig. 3-4: *Block header*

3.6 Signal format

The PLC system's signal types are allocated according to the table below. See the PLC system manual for the used for the complete address area.

3.6.1 Digital signals (ON/OFF)

The terminal can handle digital signals of the following types: For addressing Allen Bradley we refer to chapter 20, *Connecting to an Allen Bradley PLC system* and for addressing Siemens SIMATIC S5/S7 we refer to chapter 21, *Connecting to a Siemens PLC system*.

Signal	Function	A series	FX series
X	Input	●	●
Y	Output	●	●
M	Memory cell	●	●
B	Link memory cell (MELSEC NET)	●	
F	Error memory cell	●	
S	Sequence memory cell		●
T	Timer	●	●
C	Counter	●	●

Tab. 3-1: *Digital signals*

3.6.2 Analog signals (Values)

The terminal can handle analog signals of the following types: For addressing Allen Bradley we refer to chapter 20, *Connecting to an Allen Bradley PLC system* and for addressing Siemens SIMATIC S5/S7 we refer to chapter 21, *Connecting to a Siemens PLC system*.

Signal	Function	A series	FX series
D	Data register	●	●
W	Link register (MELSEC NET)	●	
R	File register	●	
T	Timer	●	●
C	Counter	●	●

Tab. 3-2: *Analog signals*

NOTE 32-bit counters (C200–C255) are not available in FX series.

3.6.3 Suffix

Double-register, rounded-off values and/or only positive values can be presented by giving an analog signal a suffix. Which signal format is permissible is shown in the PLC system manual.

NOTE

Decimal or scaled values are stored internally as floating numbers with seven digit accuracy. To avoid rounding off errors for several places of accuracy, numbers without decimals or scaling should be used. These are stored internally as whole numbers.

Format of data types

Code	Format type	Range
Dn	Simple whole number	–32768–+32767
Dn**+**	Simple (positive only)	0–+65535
Dn**L**	Double whole number	–214783648–+214783647
Dn**L+**	Double (positive only)	0–+4294967295
Dn**RB**	BCD floating number	–9999.9999–9999.9999
Dn**RF**	Floating number	±3.4E 38. Numbers larger than 1,000,000 are shown with exponent. The Decimals parameter has no function for this type of format.
Dn**I**n	Indexing	See section on Index addressing

Tab. 3-3: *Format of data types*

In the above table **n** is the number on the register, for example n=100 corresponds to D100. If D100L+ is given this means that the D100 and D101 registers are read as positive whole numbers.

3.6.4 Index addressing

Index addressing makes it possible to use the same block; for example, for the control of several motors. In the example below the torques are linked to the D100I1 register. The torque is shown for one of the three motors depending on the value in the index register, in this case the value in D0. The value in D0 decides for which motor the torque is shown. The content of the index register is added to the address and the result forms the address which is shown. Generally it can be said that:

ShownValue=content(DataRegisterAddress+content[Indexdataregister])

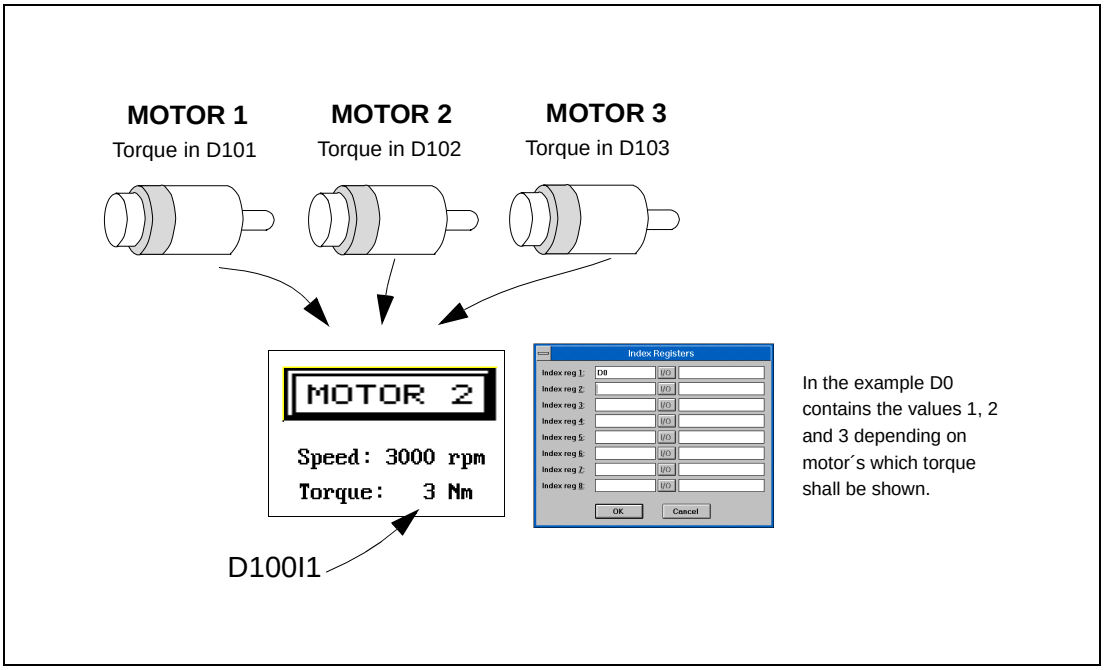


Fig. 3-5: *Index addressing*

The index register is given as a suffix to the address. It is still possible to add other suffixes after the index definition. For example, D100I1L is given for the double register. The index register will not be counted twice if the suffix L is given. I1 to I8 are linked to the data register in the **Setup** menu if the PC software is used or in the configuration mode via the terminal.

3.6.5 General functions and parameters

Colors

16 colors are available in the terminal; black, red, prussian blue, magenta, yellow, gray, light gray, light green, light red, light prussian blue, light magenta, light yellow and white.

The colors allow the creation of more realistic objects with 3-D effects and shadowing. Apart from background and foreground colors for the blocks, the colors can also be selected for the graphic objects.

The colors for the background, text and windows are determined when the terminal is configured. Colors can be chosen for example for the scales and curves for graphic objects.

Scaling

The *Offset* and *Gain* parameters are used to change the scale of the register value to a shown value according to the following equation:

DisplayValue=Offset+Gain*RegisterValue

When the object is altered from the terminal in the run-time mode, the scale is changed for the shown value according to the following equation:

RegisterValue=(DisplayValue-Offset)/Gain

Scaling does not affect the defined max and min values nor the number of decimal places.

Offset and Gain calculation

The function **OGC** is an aid for calculating the Offset and Gain parameters. Enter the max and min values for the PLC register and the max and min values for the presented value. The function will then calculate the correct value of the parameters Offset and Gain.

Reserved characters

The following ASCII characters are reserved for internal functions in the terminal and may not be used.

Hex	Dec	Character	ASCII control code
01	01		SOH
02	02		STX
03	03		ETX
04	04		EOT
05	05		ENQ
06	06		ACK
07	07		BEL
15	21	§	
23	35	# (textblock only)	

Tab. 3-4: *Reserved characters*

Positions

The number of character positions which the object will take up on the display are given by the **Position** parameter. Including comma and minus sign.

Adjustment

Whether the text is left aligned or centred is set with the **Adjust** parameter.

Frame

Whether a frame can be drawn around the object is set with the **Frame** parameter.

Maneuver

Whether the object can be altered from the terminal in the run-time mode is set with the **Maneuverable** parameter.

Min

The **Minimum value** parameter states the minimum value the object can have or the minimum limit for maneuvering.

Max

The **Maximum value** parameter states the maximum value the object can have or the maximum limit for maneuvering.

Scale division

The **Scale division** parameter states with which interval the scale numbers shall be drawn.

Scale ticks

The **Scale ticks** parameter states the interval between the scale ticks displayed.

Decimals

The **Decimals** parameter states the number of decimal places with which the value is entered.

Security level

The **Security Level** parameter states the security level for the object.
See the *Password* section.

Character size

The **Font Size** parameter states the size of the text in the object.

4 Programming using the PC software

This section describes how the terminal is programmed using the PC software. For installation instructions and more detailed information about PC software refer to the manual for the product.

All the functions can be reached with the PC software from the menu:

Fig. 4-1: *Menu*

4.1 Creating a project

A new project is created by selecting **New** in the menu **File**. The Project Settings dialog window is now shown. Here you can select the PLC system, type of terminal and any I/O name file.

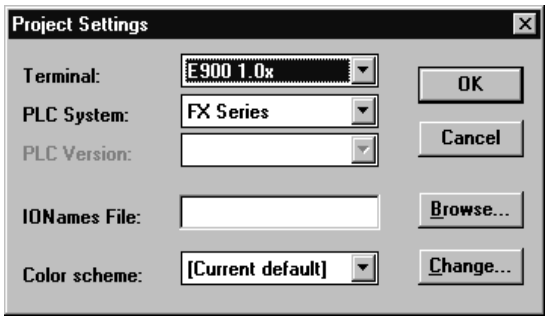


Fig. 4-2: *Project settings*

Terminal:

Select the type of terminal here.

PLC system:

Select here the PLC system to which the terminal is to be connected. The terminal can be connected to the following systems.

PLC system	Modules	Protocol
MELSEC FX	FX, FX0, FX0N, FX2N	CPU-port
MELSEC A	AnN, AnS, AnU	CPU-port, C24
MELSEC Q	QnA	CPU-port
Siemens SIMATIC S5	All	AS511 or PROFIBUS DP
Siemens SIMATIC S7	All	PROFIBUS DP
Allen Bradley	Micrologix, SLC 500, PLC 5	DF1

Tab. 4-1: *PLC system and it's protocol*

Projects in the terminal are not deleted when you change PLC system.
I/O not available in selected PLC system are not deleted.

I/O Name file:

A name file for a MELSEC MEDOC or MELSEC MEDOC *plus* project is stated here.

Color scheme:

Here you create your own color scheme and store it under a unique name. The color scheme decides the color of the background, menus, dialogs, objects etc. The color scheme is linked to MAC Programmer+ and not to the project. This means that the color scheme is valid in the whole program. When you select an object in the toolbox or in the menu the colors of the object are shown as defined in the color scheme.

4.2 Creating a block via the Block Manager

The block manager is shown when you have created a project. Which blocks are included in an application are shown here.

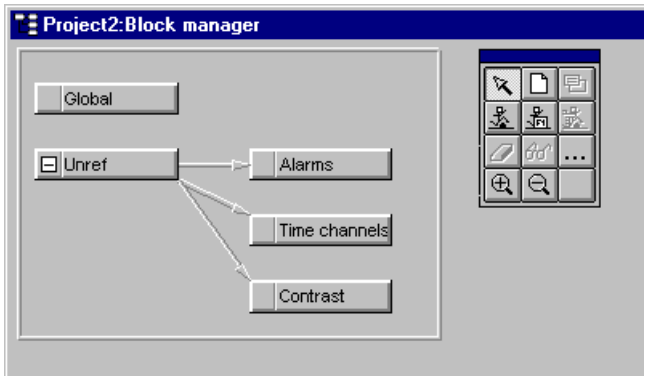


Fig. 4-3: *Block manager*

In the block manager there is a toolbox with the following functions.


Mark block and create block.


Add a new block.


Define block header for selected block.


Make a block jump to the block to be created with the cursor.


Make function key block jump to the block to be created with the cursor.


Delete selected block.


Edit selected block.


Block manager settings.


Zoom in.


Zoom out.

Define block

When you want to add a block the following dialog is shown.

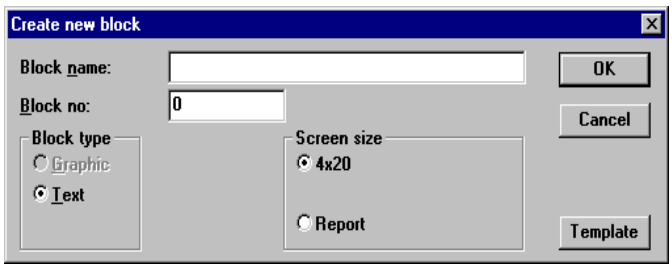


Fig. 4-4: *Create new block*

- Block name:
  A name for the block can be written here. The block name is shown in the block manager and in the block list.
- Block no.:
  The number of the block. The defined values are entered automatically if a block with assigned number already exists. The block number 0 is shown automatically on starting up and must always be used in a project.
- Block type:
  Select block type, graphic or text.
- Template:
  Here you can copy a template to the block or save the block as a template.

Define the block header

Basic parameters valid for each single block are defined in the block header. The following parameters can be defined in the block header:

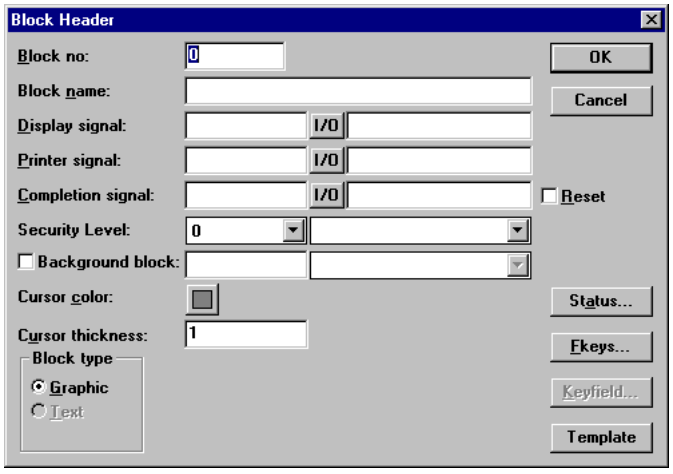


Fig. 4-5: *Block header*

- Block no.:
  The number of the block. The defined values are entered automatically if a block with assigned number already exists. The block number 0 is shown automatically on starting up and must always be used in a project.
- Block name:
  A name for the block can be written here. Block name is optional.
- Display signal:
  A digital signal which when activated results in the block being shown on the display. If display signal are used they should be ordered consecutively, for the different blocks to get the best performance. This field is not filled in if another method for changing block is used.
- Printer signal:
  A digital signal which when activated results in the block being printed out on a connected printer. The display signal and printer signal can be the same. The printer signals should be used consecutively to print out as quickly as possible.
- Completion signal:
  A digital signal which is activated when the print out is completed.
  Normally the signal is set. If the function Reset in Block Header is marked then the signal is set to zero when the print out is completed.
- Security level:
  A security level (0-8) for the block is stated here. The operator must enter a password equivalent to a given or higher security level if a security level > 0 is stated.
- Background block no:
  Here you have the possibility to load another block as a background in current block. For example if you want to use the same background in more than one block. When the graphic block editor is active you can select if the background block shall be shown in current block with the function Show Background Block in the Windows menu.
- Cursor color:
  Select the color of the cursor in graphic blocks.
- Cursor thickness:
  Select between three different alternatives.
- Block type:
  Select block type, graphic or text. It is not possible to change block type for a defined block.
- Fkeys:
  The local function keys (F1-F22) for the block are defined here. Enter a signal that will be activated when the function key is pressed. See the section on *Function Keys*.
- Template:
  Here you can copy a template to the block or save the block as a template.
- Status:
  States the display's appearance in the run-time mode. The status word has no function in the system blocks. The parameters in the status word have the following functions.
  - Cursor off:
    States if the cursor will not be visible in the block in the run-time mode.
  - Put cursor at first man. object:
    States if the cursor will be on the first maneuvrable object instead of in the top left-hand corner.
  - Disable main menu key:
    States if the [MAIN] key is to be blocked in the run-time mode when the block is shown on the display.

– Disable alarm list key:
  States if the [LIST] key is to be blocked in the run-time mode when the block is shown on the display.
– Disable PREV:
  States if the [PREV] key is to be blocked in the run-time mode when the block is shown on the display.
– MORE indication:
  States if the + sign is to be shown in the bottom right-hand corner and the top right-hand corner when there is more of the block below and above respectively than that shown on the display.
– Auto data entry:
  States if the cursor will automatically move to the next maneuvrable object after an entry. The cursor can only be placed on maneuvrable objects in this state.

4.2.1 Settings for the Block Manager

Here you make the settings for the presentation in the Block Manager.

- Block:
  Select how data should be presented in the block.
- Reference:
  Select the appearance of the Block Manager.
- Edit:
  Select special functions for the presentation in the Block Manager.

4.3 Programming blocks

Double click on current block or select Edit for marked block. The working area for the block and the tool box are now shown. The objects that can be created in the block are in the tool box.

Select an object by clicking on the object in the tool box, and place the cursor in the working area where you want the object to end up. The dialog window for selected objects is shown when you click. General parameters for the objects are described in the General section. Each type of object is explained in the sections on Graphic Presentation and Maneuvring and Text-based Presentation and Maneuvring. The object is now shown on the working area. If you have selected static text or graphics, it is written directly on the working area.

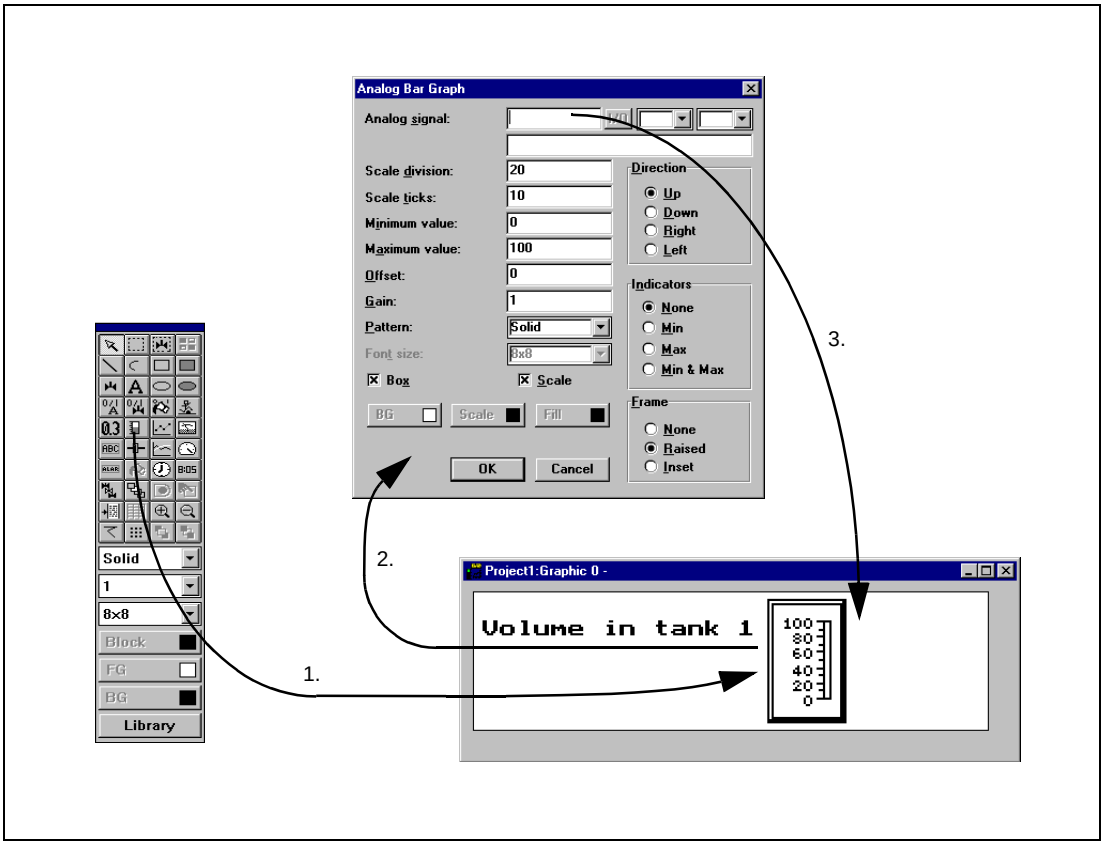


Fig. 4-6: *Programming blocks*

4.4 The File Menu

There are functions for creating, opening, saving and closing projects in the **File** menu. Printer settings, previewing print out, creating documentation headers and activating print out are also selected from here. Furthermore there are functions for testing a project and altering project settings.

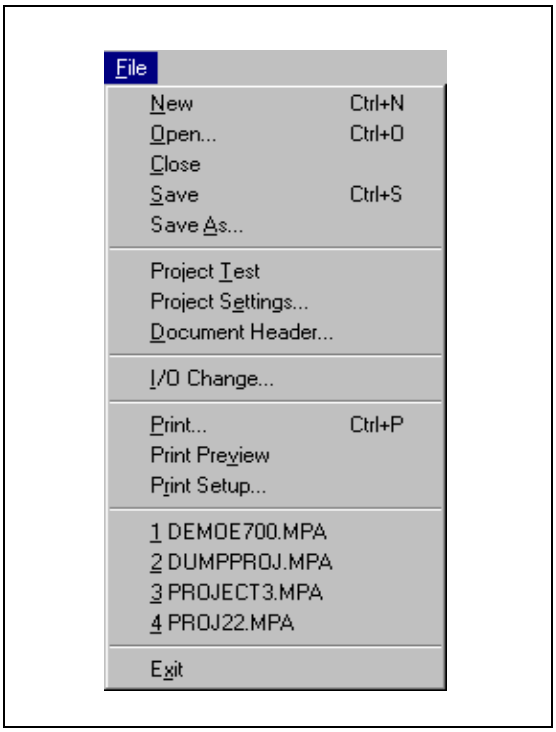


Fig. 4-7:
File menu

4.5 The Edit Menu

Functions for cutting, copying, pasting, undo and select all are found in the **Edit** menu.

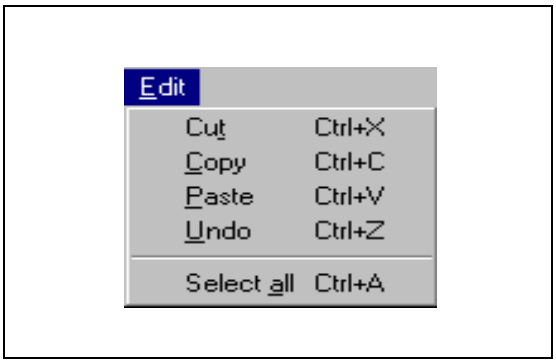


Fig. 4-8:
Edit menu

4.6 The View Menu

Function editor for block editing, symbol handling, function keys, LED, alarm, time channels, crossreference and message library are to be found in the **View** menu.

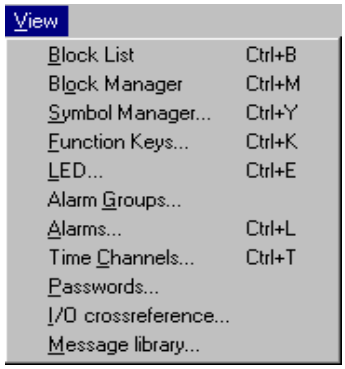


Fig. 4-9: *View menu*

Block List

Blocks included in the application are shown in the block list. In the block list you create new blocks by pressing the button **New**. Basic parameters for the block are defined in the block header.

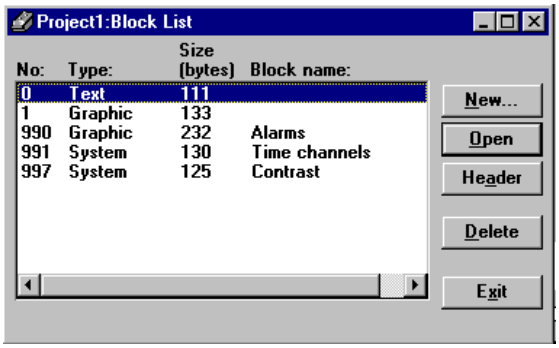


Fig. 4-10: *Block list*

Block Manager

Blocks included in the application are shown in the Block Manager graphically. In the Block Manager you create new blocks, define the block header, define block jump with the functions in the toolbox.

Symbol Manager

Activates the symbol manager. In the symbol manager you create a library of symbols in BMP format. The symbols in the symbol manager are then available in the symbol list when you create static and dynamic symbol objects. It is also possible to duplicate and edit defined symbols.

Function keys

Under Function keys you define global and local function keys. See chapter 12, *Function keys*.

LED

Under LED you define the function for the LEDs. See the chapter LEDs.

Alarm Group

Under Alarm Group you can group alarms; for example, by danger level - so that they can be more reality identified and dealt with. See chapter 7, *Alarm handling*.

Alarms

Under Alarms you define alarm messages and the signal to be connected which will activate the alarm. See chapter 7, *Alarm handling*.

Time Channels

Under Time Channels you define the time channels used to control events in the process at specified times. See chapter13, *Time control*.

Password

Under Password the password is defined for the application's different access levels. See chapter 17, *Password*.

I/O Crossreference

Under I/O Crossreference you can define several devices. These devices can be shown in an overview in the project.

Message Library

Under Message Library you define message libraries with texts in different languages or concerning different themes. See the chapter 14, *Message library*.

4.7 The Setup Menu

Functions for configuration of the terminal are to be found in the **Setup** menu.

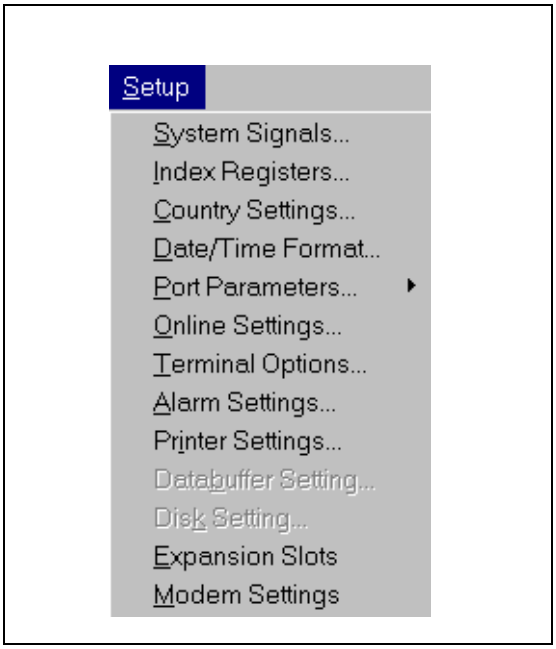


Fig. 4-11:
Setup menu

System signals

Setting of the handshaking signals between the terminal and the PLC system.

- Current display reg.:
  Data register in the PLC system which contains the number of the block shown on the display in the run-time mode. The data register is updated automatically by the terminal on changing block. This register does not affect the choice of Textblock.
- New display reg.:
  A PLC data register that determines which block will be shown on the display. D100 can be stated for example. The value 34 is entered into D100 from the PLC program or the terminal if block 34 is to be shown on the display.
- Buzzer signal:
  A register whose value decides the sound of the buzzer. See the table below. If the value i 0 then the buzzer is silent. The unit in the table is Hz.

	C	D	E	F	G	A	B
Contra	33	37	41	44	49	55	62
Large	65	73	82	87	98	110	123
Small	131	147	165	175	196	220	247
One	262	294	330	349	392	440	494
Two	523	587	659	698	784	880	988
Three	1046	1174	1318	1397	1568	1760	1975
Four	2093	2348	2636	2794	3136	3520	3950
Five	4186						

Tab. 4-2: *Buzzer signal*

- Backlight signal:
  Digital signal which determines if the display is to be switched on or off.

- No prot. ctrl. reg.:
  The first control register used in the No protocol mode. The No protocol mode is described in the *Communication* section.
- Current recipe reg:
  State the first of four registers where the terminal put the name of the last loaded recipe from the system. The name can be presented in an ASCII object. The function takes four registers (eight characters) independent of the recipe name.
- Cursor ctrl. block:
  Start register to a control block in the terminal which writes the current cursor position to the data register in the PLC system.

Register	Description
Dn0	Current cursor position X (in pixels), 0-239.
Dn1	Current cursor position Y (in pixels), 0-63.
Dn2	Status register: 0. Normal 1. User trying to move down but there is no object there. 2. User trying to move up but there is no object there. 3. User trying to move left but there is no object there. 4. User trying to move right but there is no object there.

Tab. 4-3: *Cursor control block*

- Recipe ctrl.block:
  Control block for loading/saving/deleting of recipe files via the PLC system.

Register	Content	Description
Dn0	Command	Command register set by the PLC Available commands: 0. No command. 1. Save recipe to the terminal. 2. Read recipe from the terminal. 3. Delete recipe from the terminal.
Dn1	Result code	Handshake register set by the terminal 0. Ready for new command. 1. OK 2. Recipe file write error. 3. Recipe file not found.
Dn2	File name char. 1-2	Recipe file name in the terminal.
Dn3	File name char. 3-4	
Dn4	File name char. 5-6	
Dn5	File name char. 7-8	
Dn6	Start data register	First data register to be read/written to/from recipe file.
Dn7	Number of registers	Number of registers to read/write to/from recipe file.

Tab. 4-4: *Recipe control block*

Commands

One or more of the following commands can be stated on the command line. The commands are separated by a space.

Command	Description
BR	BatchRead - communication with xxxC24, FX and A.
BSD	Disable Baud switching when communicating via transparent mode with FX2N and Q. BSD must be set in the terminal not connected to the PLC system.
DGP	Delete the alarm group from the alarm print out.
DIMn	Data register containing a value between -63 and +63 controlling the display brightness. -63 means that the display i as dark as possible and +63 means that the display is as light as possible. Normal value is 0.
ML	This command is used for indexing message libraries. State ML and a register, e.g. MLD10. The terminal will show the texts in the library addressed by the constant for the library plus the content in the register stated with the command ML.
NPn	Digital signal for switching between No protocol and Transparent/printer modes, e.g. NPM 100. Used for switching between No protocol and Transparent/printer modes in run-time mode e.g. to ring up a computer and send a message.
NTx	Timeout in ms for messages in the No protocol mode.
PDxxxxxxxx	Password that protect the transfer menu.
PRDn	Digital signal for stopping the print out, e.g. PRDM100.
PSxxxxxxxx	Password that will give access to all functions in the terminal.
RPD	RUN/PROG disable. Turns off the possibility to switch between RUN/PROG with the keys [BACKSPACE] and [MAIN]. If RPD is used you switch mode via the PC software.
Rx	Maximum number of retransmissions. x = number of attempts.
SW	Converts text with Swedish ASCII (7 bits) characters to 8 bits IBM PC extended ASCII on printing.
Tx	Timeout for x milliseconds.

Tab. 4-5: *Commands*

Index register

Index addressing of dynamic objects. For more information see chapter 3, *General*.

Country settings

- Language:
  Choice of system message language; *British English*, *German*, *Swedish* or *American English*. The default setting is English menu texts.
- Character set:
  The choice of character set. Four key codes (C1-C4) have been reserved for national characters. Choose between *Swedish*, *German*, *French*, *Spanish*, *Norwegian/Danish, Russian* and *Slavic*.

Date/Time format

Setting of date and time format.

- Date format:
  The default format can be YY-MM-DD, YYMMDD, DD.M.YY, DD/MM/YY or MM/DD/YY where Y=year, M=month and D=day.
- Time format:
  The default format can be HH:MM:SS or HH:MM where H=hour, M=minute and S=second.
- PLC register:
  Enter start address for storing date/time. Is only used if *clock →PLC* is YES.

The default setting is D9025 for the PLC system in A series.

Data register	8 higher bits	8 lower bits
D9025	year	month
D9026	day	hour
D9027	minute	second
D9028	reset	day of the week

Tab. 4-6: *Register settings*

The default setting is D8013 for the PLC system in FX and FX0N series.

Data register	Presentation
D8013	second
D8014	minute
D8015	hour
D8016	day
D8017	month
D8018	year
D8019	day of the week

Tab. 4-7: *Register settings*

- Clock used:
  Select in the list of choices. Terminal means that the terminal's built-in clock is used and PLC means that the clock in the PLC system is used.
- Clock→PLC:
  Mark if the clock in the terminal is to be transferred to the register in the PLC system.

NOTE

If the realtime clock in the PLC system is activated and the terminal clock is transferred to the same register, the clock in the PLC system is superior.

- Update interval:
  Updating interval in seconds. Only used when Clock→PLC is active.
- Daylight saving:
  The beginning and end of summer or daylight saving time is defined here. State the day of the week, week in the month, month or hour. The default setting in the PC software is selectable between Europe, the US and User defined.

Port parameters

Setting for communication with the PLC system, transparent mode, no protocol mode and printer as well as with the PC software.

Parameter	Settings
Port	RS232 or RS422
Baudrate	600–57600
Data bits	7 or 8 depending on system chosen. For commucication with the PC software the number of data bits must be 8.
Stop bits	1 or 2
Parity	Odd, even or none depending on system chosen.
Station	0. The same as the communication module or 0 if connected to CPU.

Tab. 4-8: *Port parameters*

Terminal options

Setting the terminal.

- Screen save time:
  The time in minutes after the last operation on the display when it will switch off. The default setting is 10 minutes. This function is used to optimize the display's working life.
- Key beep:
  States whether the terminal will beep when a key is pressed.
- Key repeat:
  States whether a function is to be repeated as long as the key is pressed.
- Key delay time:
  The time interval between pressing the same key before the cursor moves to the next position. Used for input of ASCII characters (A-Z etc.).
- Background:
  Select terminal background color in the list of choices by clicking BG.
- Foreground:
  Select terminal text color in the list of choices by clicking FG.
- Windows:
  Select the color of the windows shown on the display by clicking Windows.

Alarm settings

General setting of the alarm list. For more information see the section on *Alarm handling*.

- Active signal:
  Enter the digital signal which the terminal will set when an active alarm exists.
- Unack. signal:
  Enter the digital signal which the terminal will set when an unacknowledged alarm exists.
- Alarm symbol:
  Choose when the alarm symbol is to be shown on the display. In textblock ■ALARM■ is shown and in graphic block a red bell,  , in the upper right corner is shown.

Choice	Alarm symbol is shown when	
	active	unacknowledge
NO		
UNACK		●
ACTIVE	●	
ALL	●	●

Tab. 4-9: *Alarm symbols*

- List size:
  The size of the list in total kbytes. Default setting is 1.
- List erase:
  A digital signal which erases not active alarms in the alarm list when set.
- Reset:
  Marks if a signal will be reset when the list is erased.
- Backlight:
  The backlight has the following options.

  On The backlight is on while there is an alarm symbol.
  This is default.

  Off The backlight is not affected by alarms.

  Timer The backlight is turned on by new alarms and turned off when defined time has runned out.

Printer settings

Setting of page length and character for line end when printing out a textblock. For more information see the *Report print-out* section.

- Printer:
  Select printer, None or e.g. HP PLC5.
- Printer Port:
  Select the communication port connected to the printer.
- Page length:
  The number of lines which will be printed out before form feed. If the page length is set to 0 form feed never occurs. Default setting is 60.
- Paper Type:
  Select type of paper.
- Graphic orientation:
  Select the graphics orientation, portrait or landscape
- Text orientation:
  Select text orientation, portrait or landscape.
- Graphic size:
  Select size of the graphic printout.
- Newline character:
  Marks the required new line character; CR/LF, CR, LF or none.
- Handshake:
  Marks the required type of handshaking between the printer and the terminal; XON/XOFF or CTS/RTS. Refer to the printer manual for information about suitable handshaking.
- Screen dump:
  Printout of the screen. Select normal or inverted printout

Expansion ports

Settings for connected expansion cards. There are four alternatives,
IFC 128E, IFC 50E, IFC PBDP, IFC GA, IFC PI and IFC MC. For settings, see the manuals for the respective cards.

Modem Settings

See section 15.3, Modem connection.

4.8 The Block Manager Menu

The **Block manager** menu has functions for programming blocks. See section 4.3, *Programming blocks*.

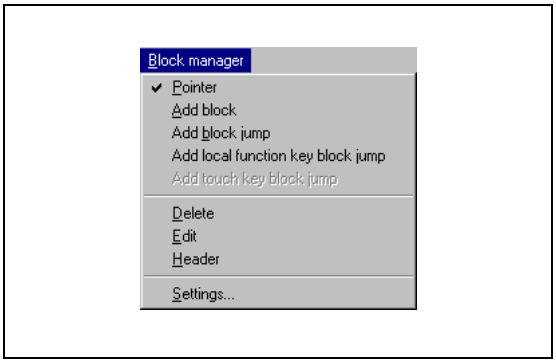


Fig. 4-12:
Block manager

4.9 The Transfer Menu

The Transfer menu has functions for the transfer of projects as well as communication settings for transferring between a PC and the terminal. See chapter 16, *Project transfer*.

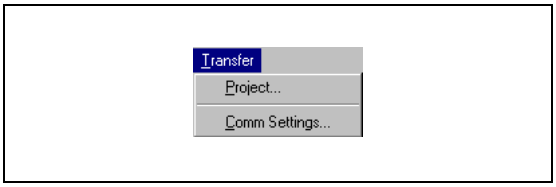


Fig. 4-13:
Transfer

NOTE The communication settings must be the same in the PC software as in the terminal.

4.10 The Windows Menu

The **Windows** menu contains standard Windows functions.

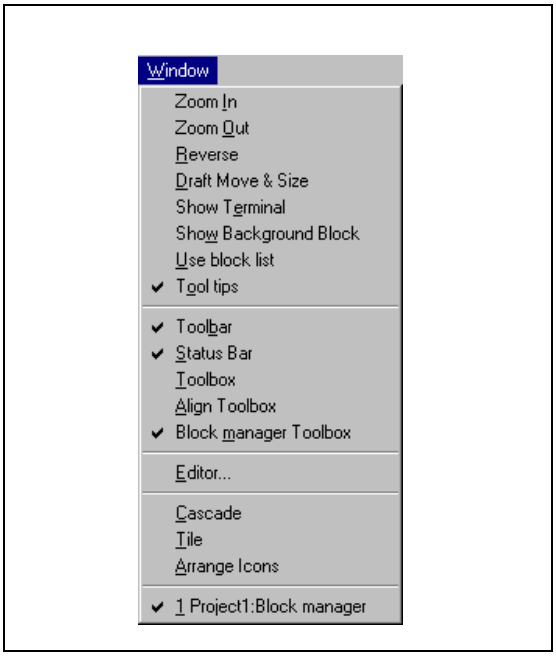


Fig. 4-14:
Windows

4.11 The Help Menu

The **Help** menu contains help texts.

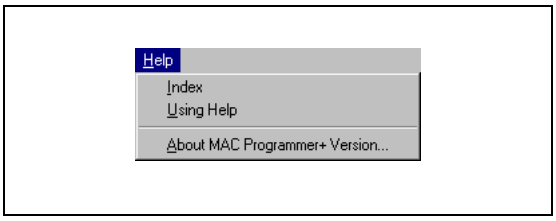


Fig. 4-15:
Help

5 Graphic presentation and maneuvering

In this section there are two examples of how to create an application, a hierarchy of menus and a sequence control. The graphic object is first presented in tabular form and each object is then described. General parameters for the objects are described in chapter 3, *General*.

5.1 Creating a hierarchy of menus

This section is an example that shows how a hierarchy of menus is built up. The application is the manufacture of film. The example shows how you can structure an application and divide it into different blocks and through the jump function create a hierarchy of menus. This is an operator-friendly way of building up an application. The operator chooses which block is to be shown depending on the state of the process. Jumps between the different blocks are controlled by the function keys, through jump objects in the block, or from the PLC system it self.

In an application built up in the form of a hierarchy of menus there is a main menu where, for example, an overview of the process is shown and from where navigation to other ”blocks” which show various details of the process.

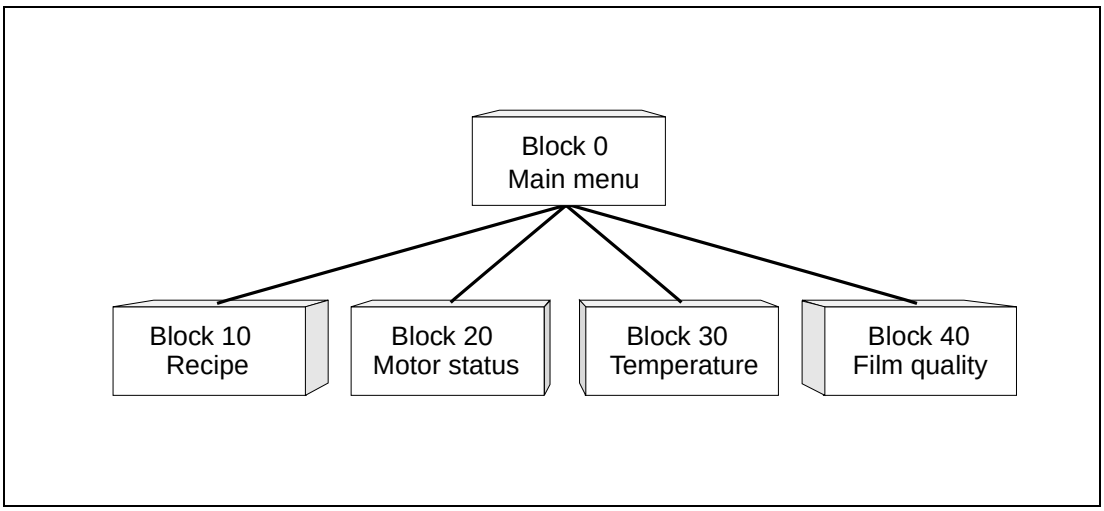


Fig. 5-1: *Application in form of a hierarchy*

Main menu

Block 0 is the main menu, a Graphic block which contains an overview of our application. There is a start/stop button as well as jump functions through the hierarchy and to the recipes, motor status, temperature and film quality. The block is constructed of static graphics and jump objects.

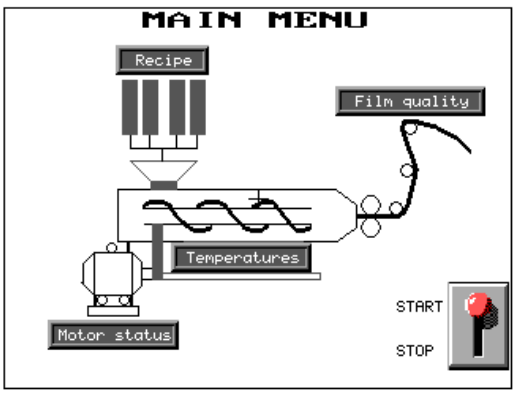


Fig. 5-2:
Main menu

The Process Method block

Process methods can be created, recalled or erased in this block. The block is constructed of potentiometer objects which can be set to required values. The objects are linked to signals in the plant which affect the various materials used in the production of film. All dynamic data in the block is saved in a recipe file with the Save recipe function. The operator can recall existing recipes by pressing the Load Recipe function. Functions for recipes are linked to the function keys F1–F3 in this example.

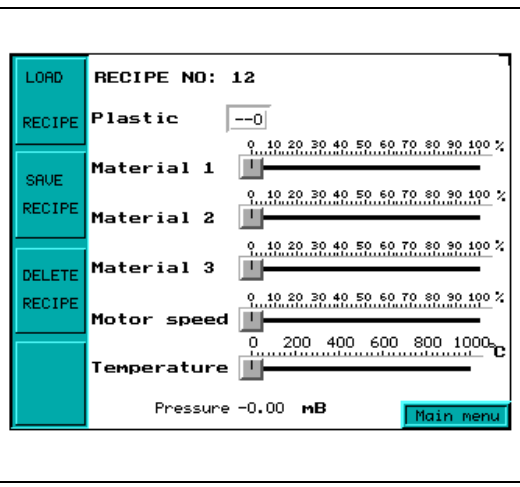


Fig. 5-3:
Process Method block

The Motor Status block

There are two VU meter objects and a bar graph object. The VU meters are linked to the signals for speed and current. The bar graph object is linked to the signal for temperature.

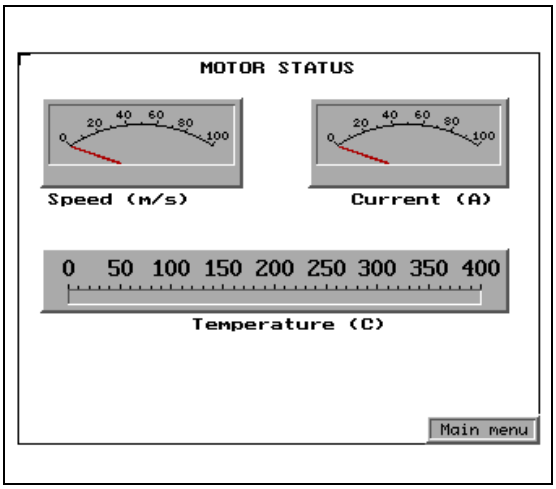


Fig. 5-4:
Motor Status block

The Temperature block

There are three bar objects in the Temperature block which present the extruder temperatures in the zones 1–3. The object is linked to the respective functions in our application.

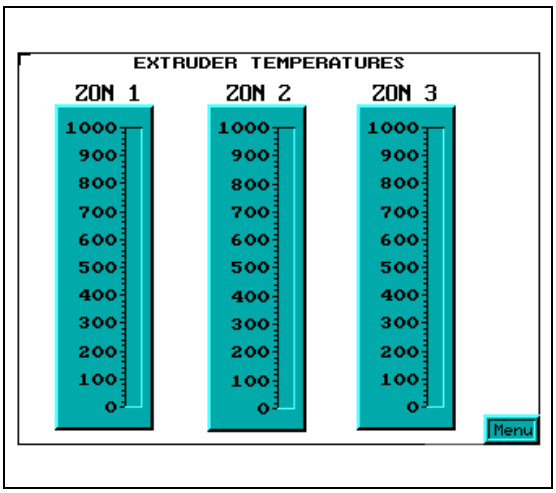


Fig. 5-5:
Temperature block

The Film Quality block

The quality of the film manufactured is presented in the Film Quality block. Furthermore there is a jump object for jumping to the main menu.

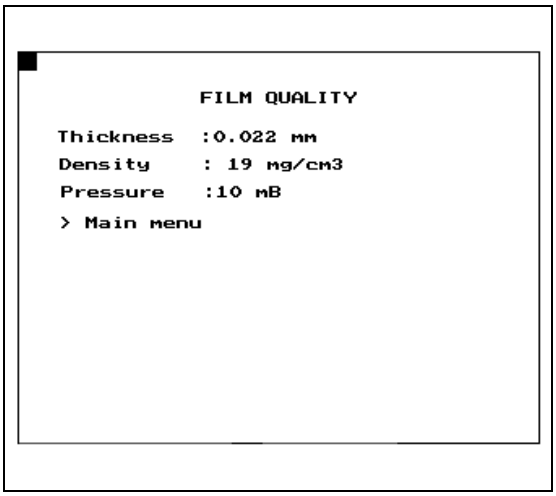


Fig. 5-6:
Film quality

Creating a sequence control

This section is an example that shows how a simple sequence control is constructed. The application is the monitoring of a washing machine.
The PLC system controls which block is to be shown.

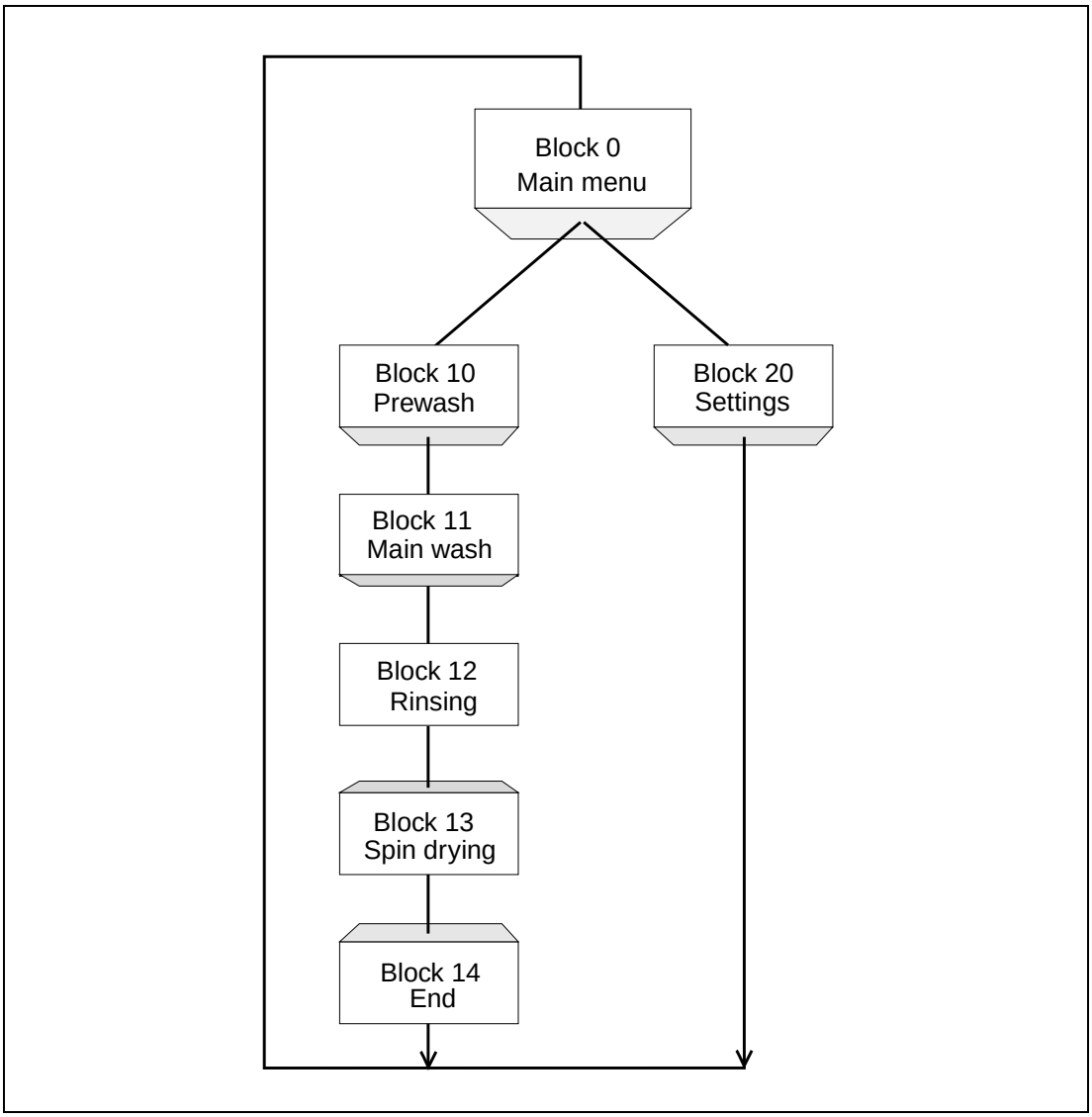


Fig. 5-7: *Sequence control*

Main menu

Block 0 is the main menu, a Graphic block which shows a monitoring picture of what happens during washing. There is a start/stop button as well as dynamic information about program choice, temperature, wash time and time remaining. Any alarms are shown in the alarm line object.
Furthermore there is a jump object for jumping to a block where you can select wash program.

Fig. 5-8:
Main menu

The Prewash, Main Wash, Rinse and Spin Drying blocks

The Graphic blocks contain static graphics in the form of a washing machine as well as an ASCII object which shows the status and an analog text object which shows the time remaining. The blocks for Main Wash, Rinse and Spin Drying are constructed in the same way. Which of these blocks is shown on the display is determined by the PLC system.

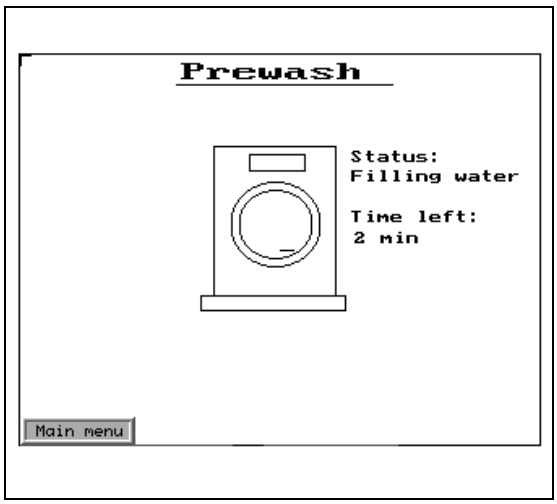


Fig. 5-9:
Prewash, Main Wash, Rinse and Spin Drying

The End of Program block

The Graphic block contains the text End of Program as well as a jump object to jump back to the main menu.

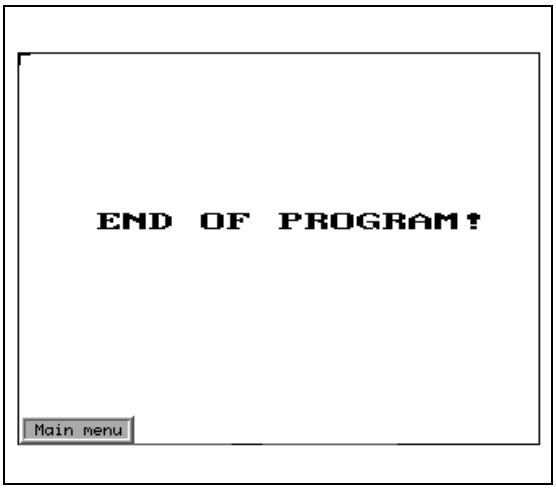


Fig. 5-10:
End of program

5.2 Graphic objects

Static graphic objects

Static graphic objects are used to draw graphics which are not affected by the process.

Symbol	Object
	Arc
	Ellipse, filled
	Line
	Polygon line
	Rectangle, filled, 3-D
	Key field object
	Symbol
	Text

Tab. 5-1: *Overview of static graphic objects*

NOTE Static object will always be drawn behind dynamic objects in run-time mode.

Digital graphic objects

Digital graphic objects are linked to digital signals in the PLC system.

Symbol	Object	Description
	Digital symbol	Switch between two symbols depending on the state of a digital signal.
	Digital text	Switch between two texts depending on the state of a digital signal.
	Digital fill	Used to fill in a framed area depending on the state of a digital signal.

Tab. 5-2: *Overview of digital graphic objects*

NOTE It is not allowed to place several dynamic objects on top of each other.

Analog graphic objects

Analog graphic objects are linked to the registers in the PLC system.

Symbol	Object	Description
	ASCII	Handle ASCII strings in Graphic blocks.
	Analog fill	Used to fill in a framed area in one of sixteen colors depending on the value of a register.
	Diagram	Used to draw X/Y diagrams based on the content of the PLC registers.
	Multichoice	Linked to a register which can assume up to eight different states. Text of up to 30 characters can be linked to each state.
	Multisymbol	Shows one of up to eight symbols depending on the value in the data register.
	Numeric	Entering and presentation of values as numbers
	Slide	Makes possible the increasing and decreasing the value of an analog signal.
	S meter	Creates a graphic speedometer on the display.
	Bar	Presents values in as a bar chart.
	Trend	Presents values collected from the registers as a curve.
	VU meter	Creates a graphic VU meter on the display.
	Message	Used to control which text in a Message library to be shown.
	Analog numeric table	Creates a table with numeric objects.

Tab. 5-3: *Overview of analog graphic objects*

NOTE It is not allowed to place several dynamic objects on top of each other.

Other objects

Symbol	Object	Description
	Jump	Jump to another block.
	Alarm banner	Used to show a line in the alarm list.
	Digital clock	Object to show a digital clock.
	Analog clock	Object to show an analog clock.

Tab. 5-4: *Overview of other objects*

NOTE Parameters which apply generally for object are described in the *General* section.

Digital symbol

Object which is used to switch between two chosen symbols depending on the state of a digital signal.

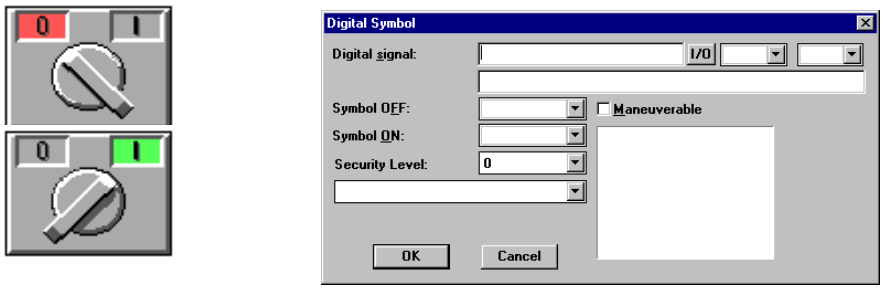


Fig. 5-11: *Dialog box digital symbol*

Parameter	Description
Digital signal	The address of the signal.
Symbol OFF	Select the symbol which is to be shown when the signal is 0.
Symbol ON	Select the symbol which is to be shown when the signal is 1.
Maneuverable	When selected the object can be forced from the terminal in run mode.
Security Level	Select security level for the object. 0 means no security level.

Tab. 5-5: *Parameters for digital symbols*

Digital text

Text object which is used to switch between two entered texts depending on the state of a digital signal. The maximum text length is 30 characters.

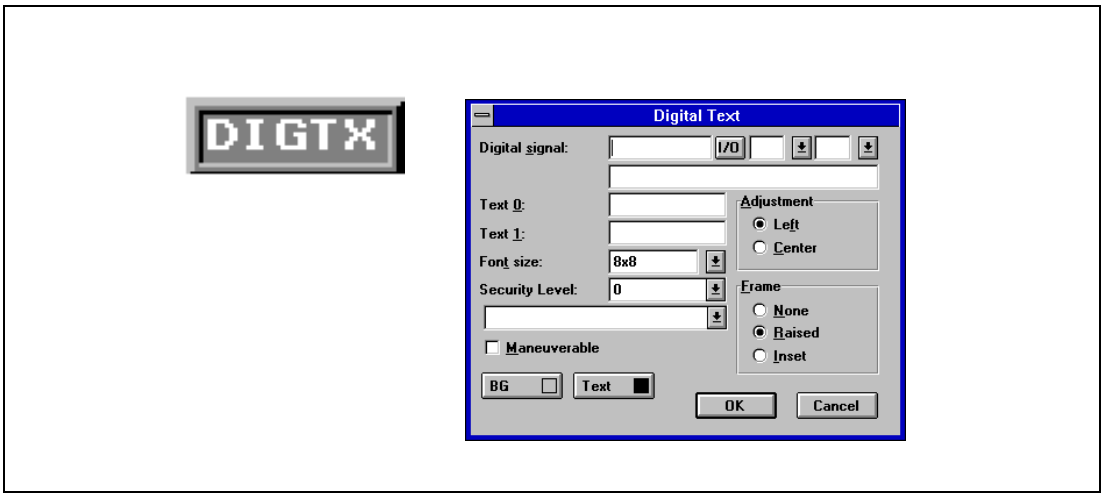


Fig. 5-12: *Dialog box digital text*

Parameter	Description
Digital signal	The address of the digital signal.
Text 0	The text which is to be shown when the signal is 0.
Text 1	The text which is to be shown when the signal is 1.
Font size	Select font size.
Security Level	Select security level. 0 means no security level.
Maneuverable	When selected the object can be forced from the terminal in run mode.
Adjustment	Select whether the text is to be left aligned or centred.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color of the object.
Text	Select text color in the object.

Tab. 5-6: *Parameters for digital texts*

Digital fill

Objects used for filling framed areas with a selected color.

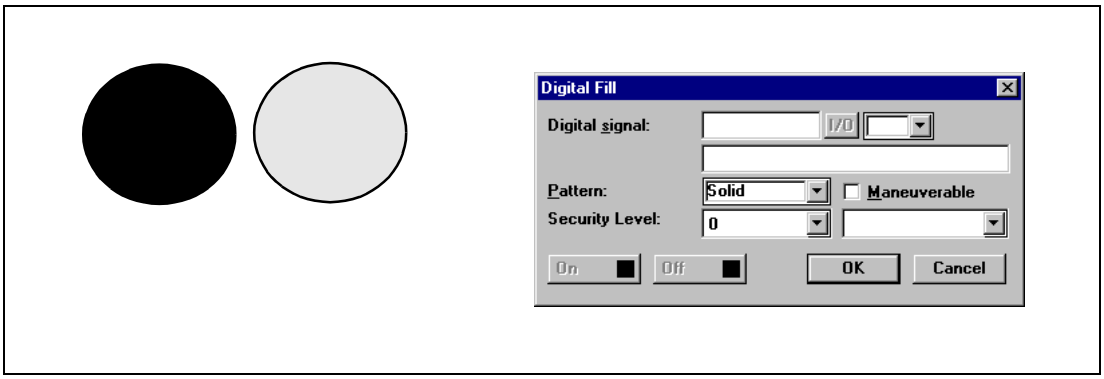


Fig. 5-13: *Dialog box digital fill*

Parameter	Description
Digital signal	The address of the signal.
Symbol OFF	Select the symbol which is to be shown when the signal is 0.
Symbol ON	Select the symbol which is to be shown when the signal is 1.
Maneuvrable	When selected the object can be forced from the terminal in run mode.
Security Level	Select security level for the object. 0 means no security level.

Tab. 5-7: *Parameters for digital symbols*

NOTE

Filling areas that are too irregular can cause system failure in run-time mode. Filling can, in certain cases, make the screen slow.

Placing an object

The program calculates what is to be filled in the specified area. It´s therefore important that the object is correctly placed. The filled area is limited only by static objects and static parts of dynamic objects.

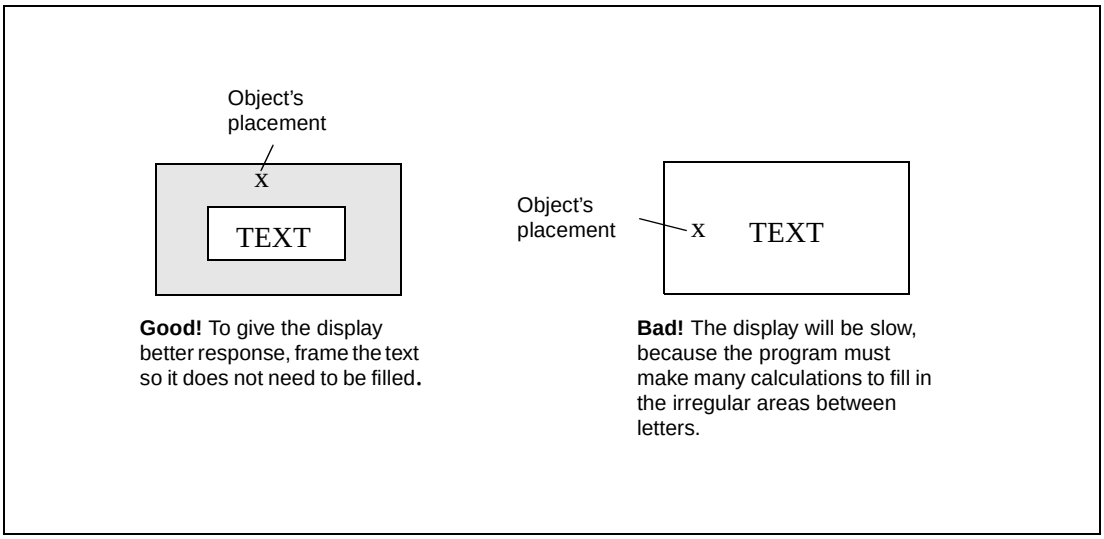


Fig. 5-14: *Placing an object*

ASCII

Object for handling ASCII strings in Graphic blocks. ASCII objects provide the opportunity to present texts stored in the CPU's data register.
The texts should be in IBM extended ASCII format. The text is converted from 8-bit IBM extended ASCII to Swedish 7-bits ASCII if SW is stated on the command line in System Signals.

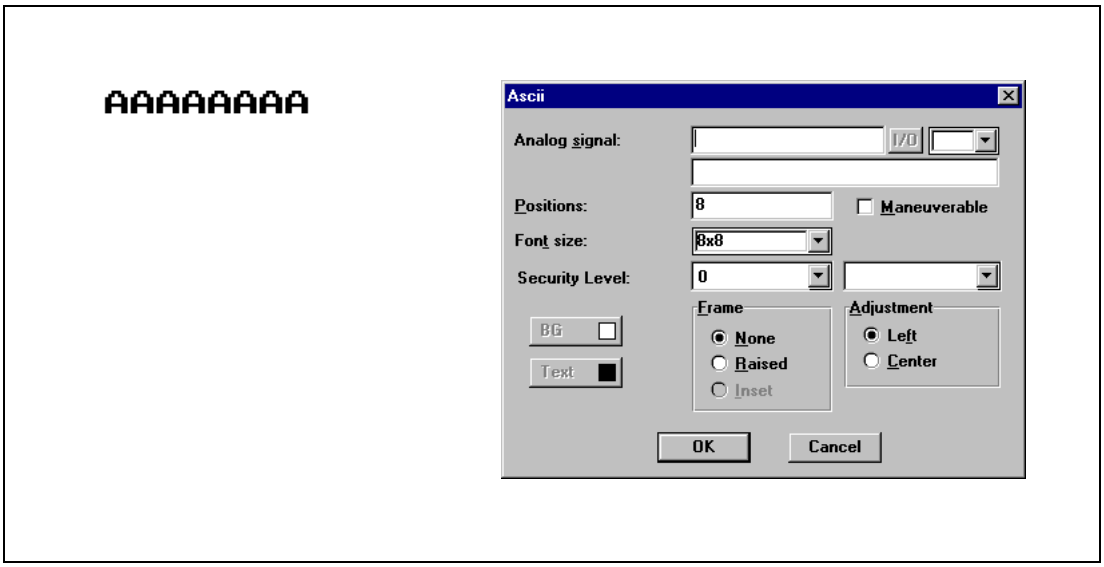


Fig. 5-15: *Dialog box ASCII object*

Parameter	Description
Analog signal	State the register where the text for the first position is to be stored.
Positions	State the number of positions the text will occupy on the display.
Font size	Select font size.
Security Level	Select security level for the object.
Maneuvrable	When selected the object can be forced from the terminal in run mode.
Adjustment	Select whether the object is to be left aligned or centred.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color of the object.
Text	Select the color of the text in the object.

Tab. 5-8: *Parameters for ASCII objects*

Analog fill

Objects used for filling framed areas with one of sixteen colors depending on the value in a register. The colors are used according to the following table.

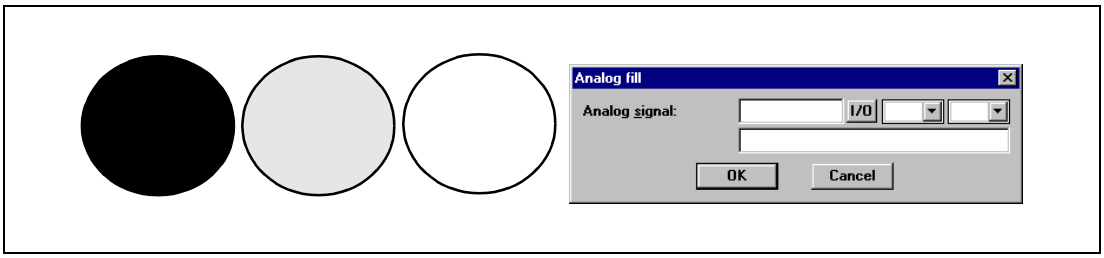


Fig. 5-16: *Dialog box Analog fill*

Parameter	Description
Analog signal	State the register whose value decides the color of the object. See the following table.

Tab. 5-9: *Parameter*

Register value	Color	Register value	Color
0	Black	8	Grey
1	Blue	9	Light blue
2	Green	10	Light green
3	Cyan	11	Light cyan
4	Red	12	Light red
5	Magenta	13	Light magenta
6	Yellow	14	Light yellow
7	Light grey	15	White

Tab. 5-10: *Parameter setting*

For limitations and how to place an object see *Digital fill*.

Diagram

Objects used to draw X/Y diagrams or bar graphs based on the content of the PLC register. This is a real-time function. The object is normally used for non-time-based presentations. Time-based display with a sample time of <1 second can be shown if the PLC creates data collection. As an example we shall say the value in D100 is the first X-coordinate and the value in D200 the first Y-coordinate. The number of register pairs will be 4.
The table and the figure below illustrate our example.

X-coordinate	Register	Value	Y-coordinate	Register	Value
X0	D100	0	Y0	D200	11
X1	D101	41	Y1	D201	40
X2	D102	51	Y2	D202	85
X3	D103	92	Y3	D203	62

Tab. 5-11: *Coordinates*

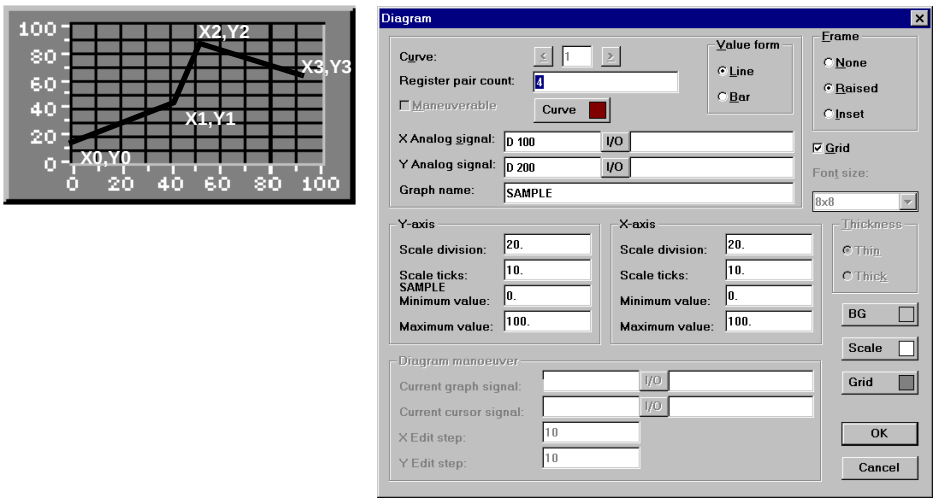


Fig. 5-17: *Diagram*

Parameter	Description
Curve	Select curve 1 or 2.
Register pair count 1-2	The number of register pair to be drawn as dots or bars for the curve respective.
Y Analog signal 1-2	The PLC register which contains the first Y-coordinate for the curve respective.
X Analog signal 1-2	The PLC register which contains the first X-coordinate for the curve respective.
Maneuverable 1-2	Select if the curve respective can be altered from the terminal in the run-time mode.
Curve 1-2	Select the color of the curve respective.
Value form 1-2	Select whether the diagram shall be a bar graph or linear graph for the curve respective. A bar is drawn for each register pair in the bar graph. The X/Y coordinates are marked in as dots between which a line is drawn in a linear graph.
Y Scale division	The interval between the scale number on the Y-axis.
Y Scale ticks	The interval between the scale ticks displayed on the Y-axis.
Y Minimum value	The minimum value of the Y-coordinate.
Y Maximum value	The maximum value of the Y-coordinate.
X Scale division	The interval between the scale number on the X-axis.
X Scale ticks	The interval between the scale ticks displayed on the X-axis.
X Minimum value	The minimum value of the X-coordinate.
X Maximum value	The maximum value of the X-coordinate.
Current graph signal	The register which contents decides what curve to be edited in run-time mode.
Current cursor signal	The register which contents decides which point on the curve to be edited in run-time mode.
X Edit step	The size of the step when you press an arrow key in run-time.
Y Edit step	The size of the step when you press an arrow key in run-time.
Frame	Select if a frame shall be drawn around the object.
Thickness	Select the thickness of the curves.
BG	Select background color of the object.
Scale	Select the color of the scale for the object.
Grid color	Select the color of the grid in the diagram.
Grid	When selected the grid is drawn in the diagram.

Tab. 5-12: *Parameter settings for diagrams*

Multiple choice

Objects that can have several states. The object is linked to a register which can assume a maximum of eight states. Each state can be allocated a text, maximum 30 characters long.

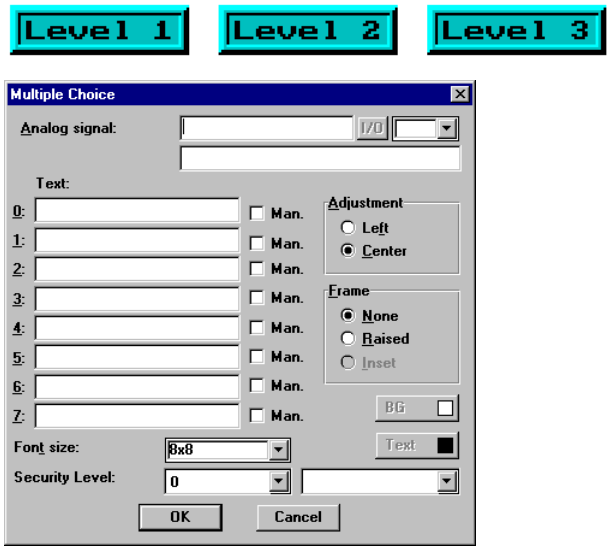


Fig. 5-18: *Dialog box multiple choice object*

Parameter	Description
Analog signal	The register that decides which text that is to be displayed.
Text 0-7	The texts which are to be shown for each state of the object.
Man 0-7	When selected the object can be maneuvered to this state in run-time mode. See the section Run-time mode for maneuvering.
Adjustment	Select whether the text shall be left aligned or centred.
Frame	Select if a frame shall be drawn around the object.
Font size	Select font size in the object.
Security Level	Select security level for the object.
BG	Select background color for the object.
Text	Select the color of the text in the object.

Tab. 5-13: *Parameter settings for multiple choice objects*

Multisymbol

Objects that can show one of a maximum of eight different symbols depending on the value in a data register. The PLC register can also control the symbol's location on the display.

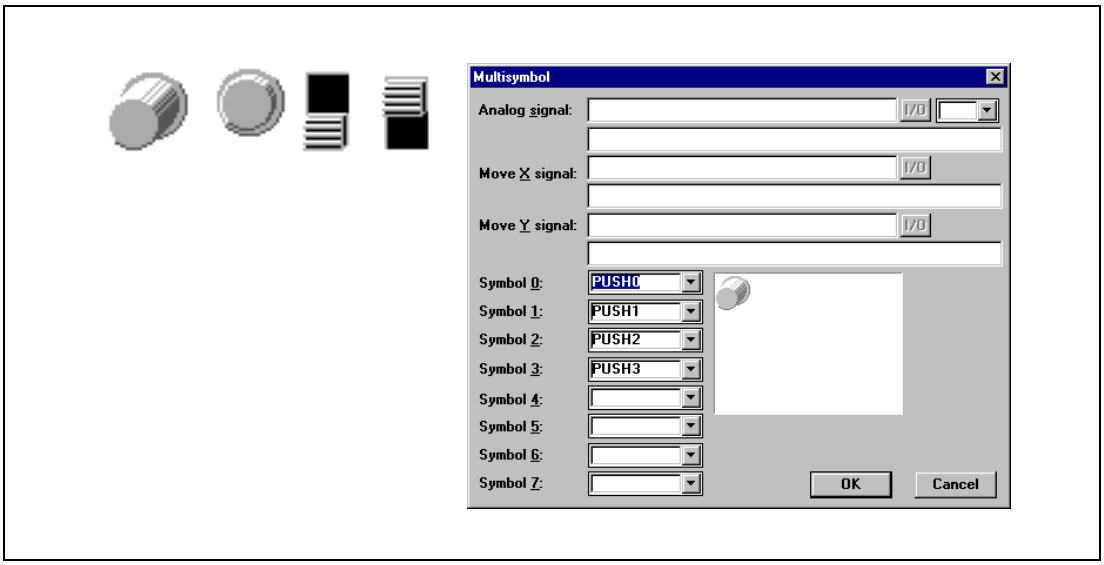


Fig. 5-19: *Dialogbox multisymbol*

Parameter	Description
Analog signal	The register that decides which symbol to be displayed. If the register contains the value 1 symbol 1 is displayed etc.
Move X	The value which contains the X-coordinate for the object. Permissible values are 0-639.
Move Y	The value which contains the Y-coordinate for the object. Permissible values are 0-479.
Symbol 0-7	Select the symbol to be displayed. If the value of the register is 0, then symbol 0, is displayed etc.

Tab. 5-14: *Parameter settings for multisymbols*

NOTE

Move X and Y only need to be stated if the object is to be moved around on the display with the help of the PLC.

Numeric

Objects for entry and presentation of values in numerical form. It is used for example to create entry fields.

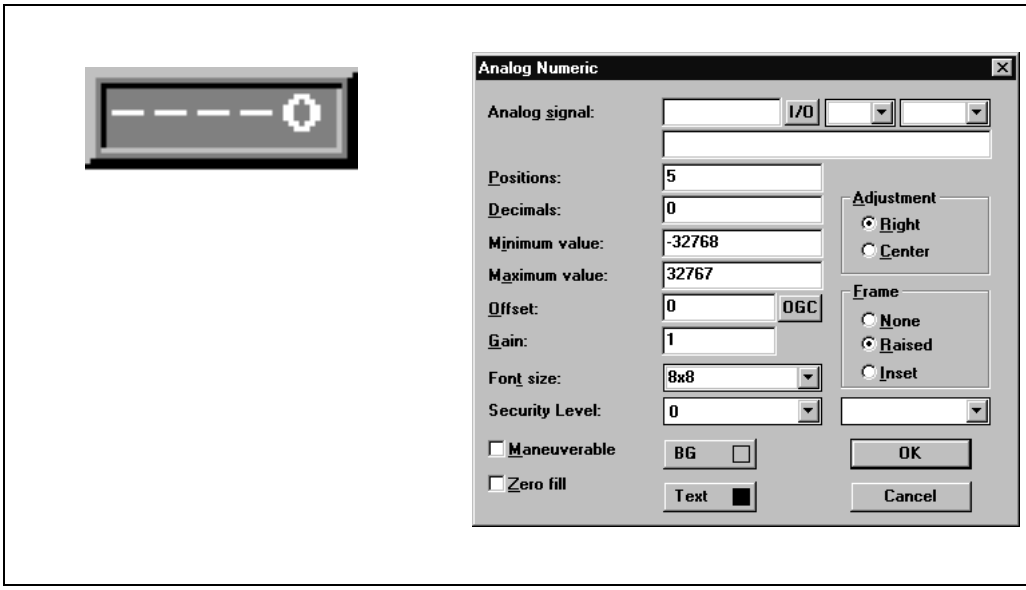


Fig. 5-20: *Dialog box numeric*

Parameter	Description
Analog signal	The address of the signal. See the chapter *General*.
Positions	The number of positions the entered value will take up.
Decimals	The number of decimals the value shall be entered with.
Zero fill	Select if zeros are to be printed out in empty positions.
Adjustment	Select if the input field shall be left aligned or centred.
Maneuverable	Select if the object can be altered from the terminal in the run-time mode.
Minimum value	The minimum limit for maneuvering.
Maximum value	The maximum limit for maneuvering.
Offset and **Gain**	Used to scale register value. See the chapter *General*.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color for the object.
Text	Select the color of the text in the object.

Tab. 5-15: *Parameter settings for numeric objects*

NOTE Minimum value and maximum value are only active when maneuvering the object.

Slide

Objects which make it possible to increase or decrease the value of an analog signal.

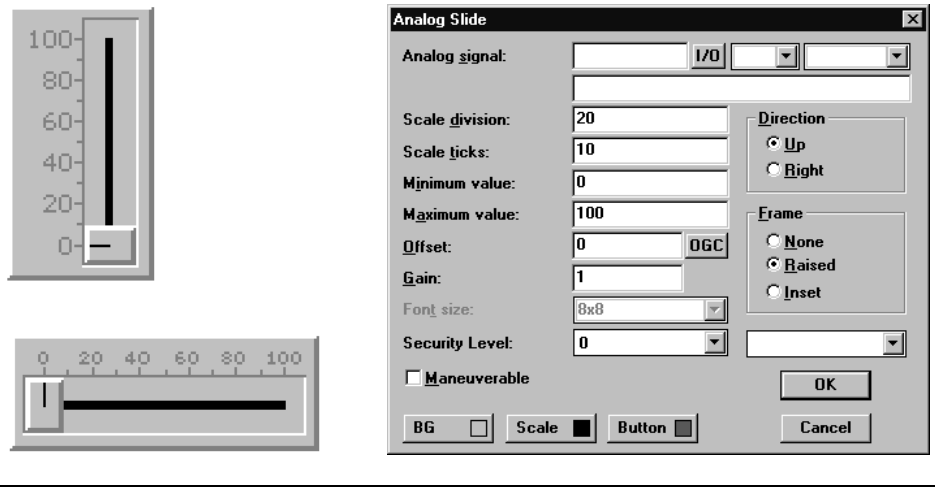


Fig. 5-21: *Dialog box slide*

Parameter	Description
Analog signal	The signal address. See the chapter *General*.
Minimum value	The minimum value the object can have.
Maximum value	The maximum value the object can have.
Direction	Select whether the object shall be presented vertically or to the right.
Security level	Select security level for the object.
Maneuverable	Select whether the object can be altered from the terminal in the run-time mode.
Scale division	State the interval between the scale numbers to be drawn.
Scale ticks	State the interval between the scale ticks displayed.
Offset and **Gain**	Used to scale register value. See the chapter *General*.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color for the object.
Scale	Select color of the scale in the object.
Button	Select color of the indication button in the object.

Tab. 5-16: *Parameter settings for slides*

NOTE The suffixes RB and RF do not work for potentiometer objects.

S-meter

Objects for creating a graphic speedometer on the display.

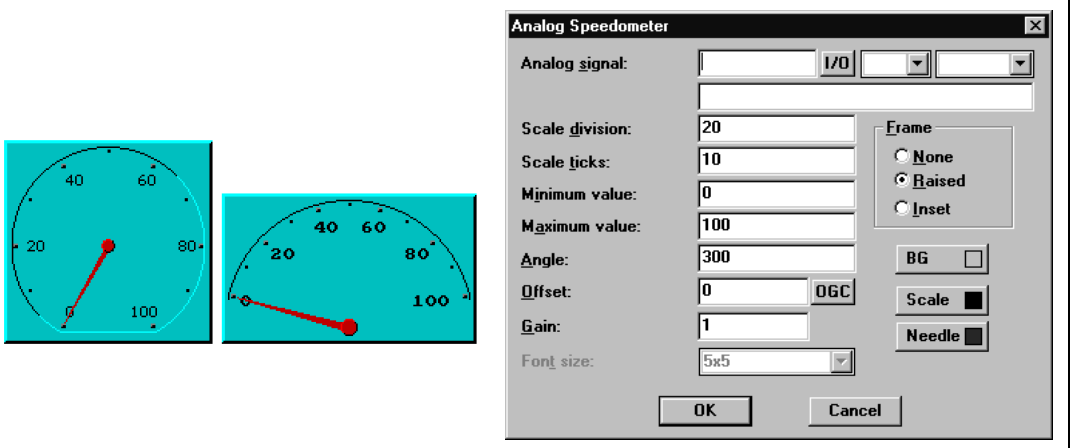


Fig. 5-22: *Dialog box S-meter*

Parameter	Description
Analog signal	The address of the signal. See the chapter *General*.
Minimum value	The minimum value the object can display.
Maximum value	The maximum value the object can display.
Scale division	State the interval between the scale numbers to be drawn.
Scale ticks	State the interval between the scale ticks displayed.
Angle	State the angle, working area, for the object. 10-360 degrees.
Offset and **Gain**	Used to scale register value. See the chapter *General*.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color for the object.
Scale	Select the color of the scale in the object.
Needle	Select the color of the needle in the object.

Tab. 5-17: *Parameter settings for S-meters*

Bar

Objects which present whole or decimal numbers in the form of a bar graph.

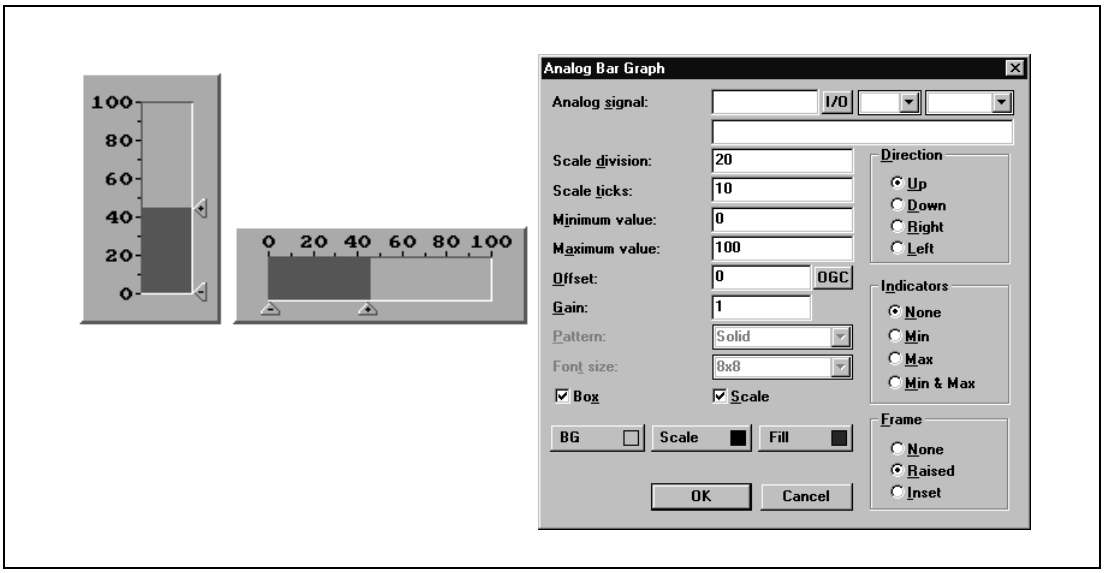


Fig. 5-23: *Dialog box bar*

Parameter	Description
Analog signal	The address of the signal. See the chapter *General*.
Minimum value	The minimum value the signal can have.
Maximum value	The maximum value the signal can have.
Direction	Select whether the bar is to be shown to the right, left, up or down.
Scale	Select whether the scale is to be shown.
Box	Select whether a box is to be drawn around the bar.
Indicators	Select whether the highest and/or the lowest value for the signal shall be shown on the axis. Indicators are set to zero when the terminal is switched on. Zeroing can also be carried out in the run-time mode, see the Run-time mode section. The indicators only function for 16-bit numbers.
Scale division	State the interval between the scale numbers to be drawn.
Scale ticks	State the interval between the scale ticks displayed.
Fill	Select fill color.
Offset and **Gain**	Used to scale register value. See the chapter *General*.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color for the object.
Scale	Select the color of the scale in the object.

Tab. 5-18: *Parameter settings for bars*

Trend

Objects which graphically present collected values from analog signals.

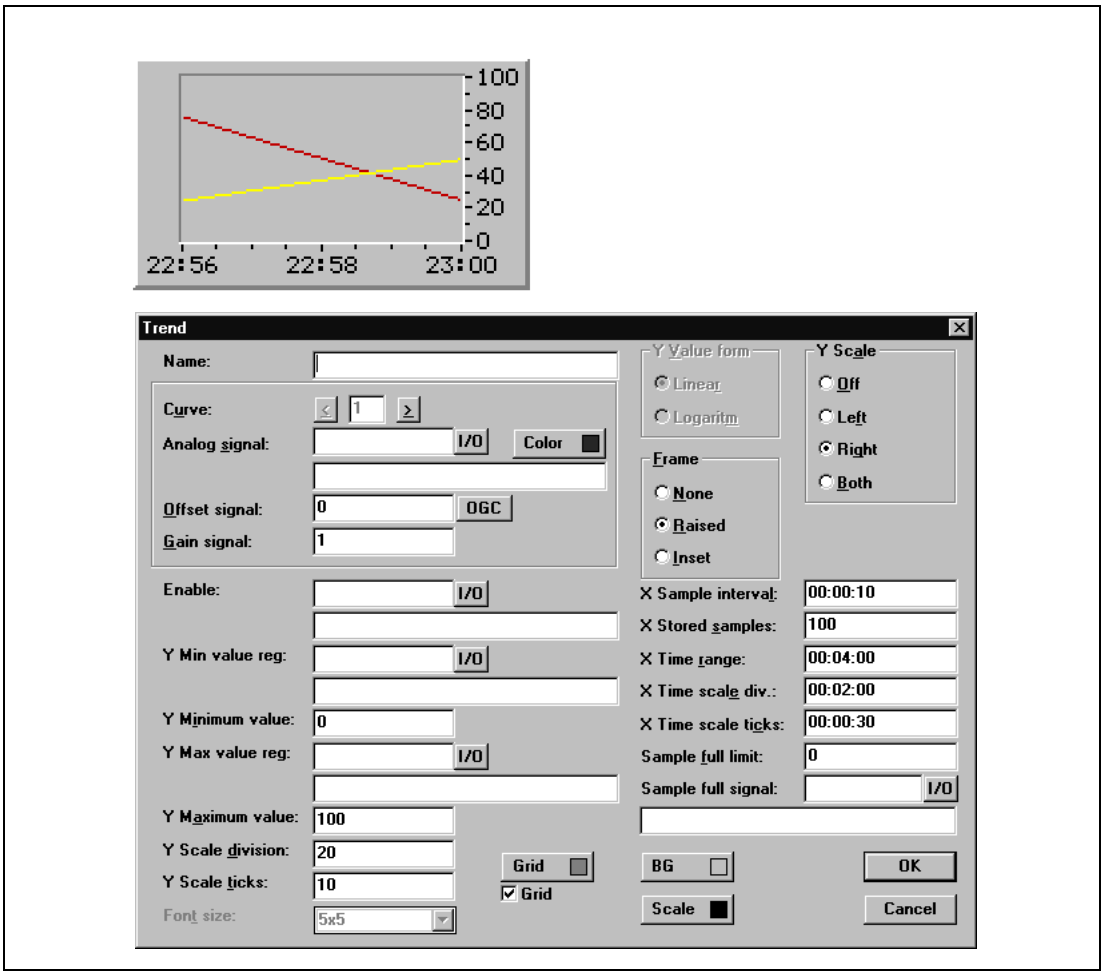


Fig. 5-24: *Dialog box trend*

Parameter	Description
Name	State the name of the object. The name must be unique for each object.
Curve	Select the curve you want to edit (1-6).
Analog signal	The analog signals the object shall sample and present the values for. Only 16-bit numbers can be used.
Color	Select a color for each curve.
Offset and Gain	Used to scale register value. See the chapter *General*.
Enable	The digital signal which when set starts the collection of data. Data collection ceases if the signal is set to zero. The parameter needs not be stated.
Y Min value reg.	The minimum value on the Y-axis loaded from the stated PLC register.
Y Minimum value	The minimum value on the Y-axis. (Ignored if Y Min value reg. is defined.
Y Max value reg.	The maximum value on the Y-axis loaded from the stated PLC register.
Y Maximum value	The maximum value on the Y-axis. (Ignored if Y Max value reg. is defined.)
Y Scale	States which side of the Y-axis shall be shown, to the left, right, both or neither.
Y Scale division	The interval between the scale number on the Y-axis.
Y Scale ticks	The interval between the scale ticks displayed.
X Sample interval	States the time interval between the collection of data. Minimum value is 1 second.
X Stored samples	States how many values shall be stored. The maximum number of values are 4800.
X Time range	States the time interval to be shown in the trend diagram.
X Time scale division	The interval between the scale number on the X-axis.
X Time scale ticks	The interval between the scale ticks displayed.
Sample full limit	State the number of samples when the samples full signal shall be set.
Sample full signal	Select a signal which will be set when the number of samples reach the value stated in the parameter Sample full limit.
Frame	Select if a frame shall be printed around the object.
BG	Select background color for the object.
Scale	Select the color of the scale in the object.
Grid	Select if grid shall be displayed in the object.
Grid color	Select a suitable color for the grid.

Tab. 5-19: *Parameter settings for trends*

VU-meter

Object for creating a graphic VU meter on the display.

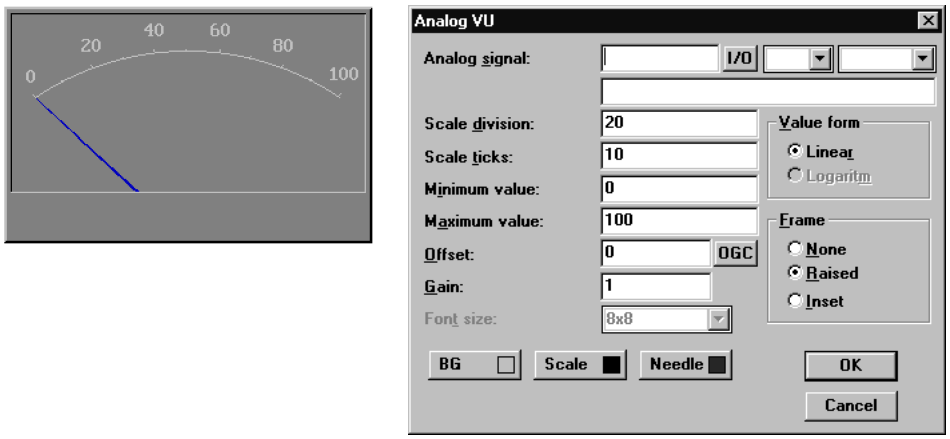


Fig. 5-25: *Dialog box VU-meter*

Parameter	Description
Analog signal	The address of the signal. See the chapter *General*.
Minimum value	The minimum value the signal can have.
Maximum value	The maximum value the signal can have.
Scale division	State the interval between the scale numbers to be drawn.
Scale ticks	State the interval between the scale ticks displayed.
Offset and **Gain**	Used to scale register value. See the chapter *General*.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color for the object.
Scale	Select the color of the scale in the object.
Needle	Select the color of the needle in the object.

Tab. 5-20: *Parameter setting for VU-meters*

Jump

Object for jumping to another block. Makes it possible to build up a hierarchy of menus in the application. Jumping back to the previous block is done by pressing the [PREV] key (up to nine levels back). See chapter 12, *Function Keys*.

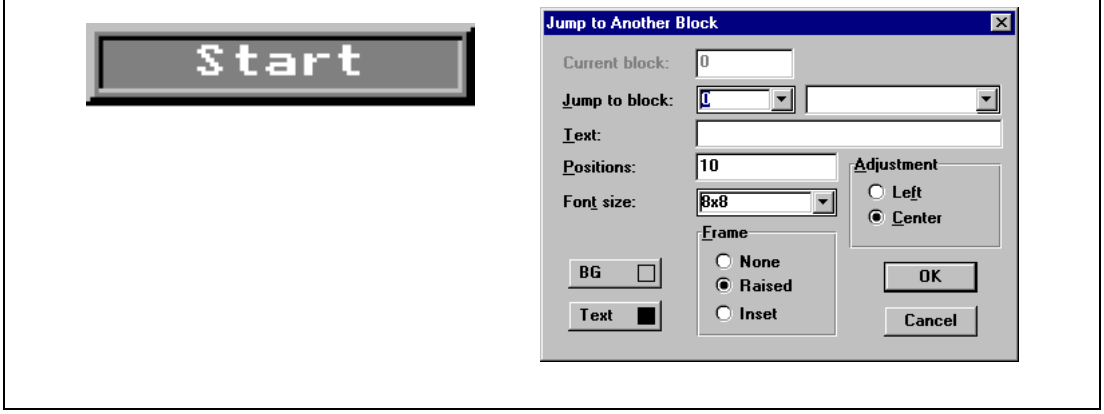


Fig. 5-26: *Dialog box jump*

Parameter	Description
Jump to block	States the number of the block to which the jump is to be made.
Positions	The number of positions the text will take up on the display.
Text	Enter any text that will be displayed.
Adjustment	Select whether the text shall be left aligned or centred.
Frame	Select if a frame shall be drawn around the object.
Font size	Select font size.
BG	Select background color for the object.
Text	Select the color of the text in the object.

Tab. 5-21: *Parameter settings for jumps*

NOTE If a jump is generated to a block that does not exist in run mode an error message is shown.

Alarm banner

Object used to show a line in the alarm list.

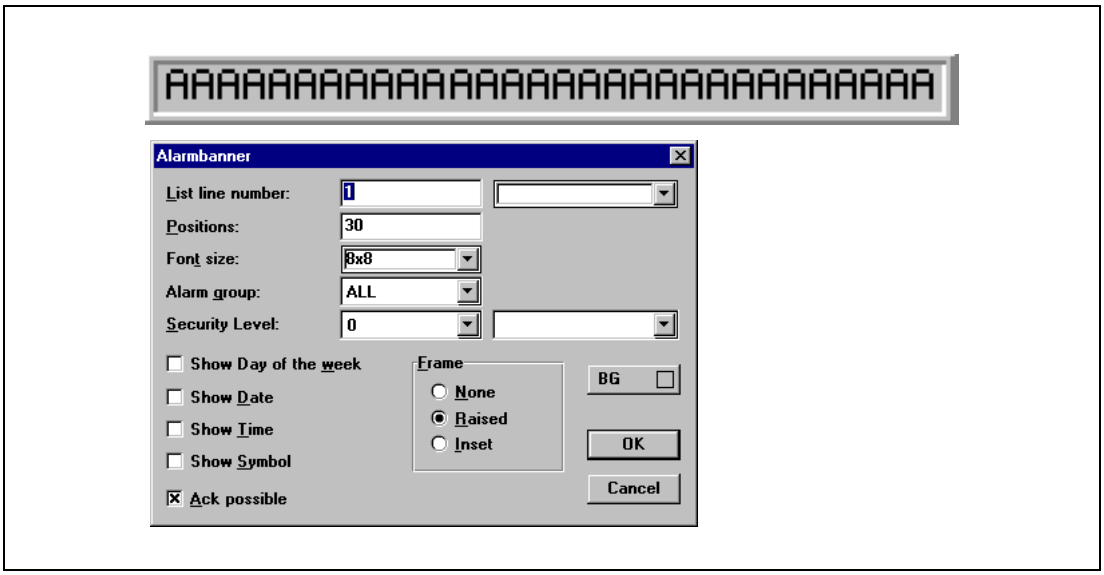


Fig. 5-27: *Dialog box alarm banner*

Parameter	Description
List line number	States the number of the line in the alarm list from which information is to be collected (1=first line, 2=second line etc.).
Positions	The number of positions to be displayed.
Font size	Select font size for the text in the object.
Show day of the week	Select whether the day of the week will be shown in the alarm line.
Show date	Select whether the date will be shown in the alarm line.
Show time	Select whether the time will be shown in the alarm line.
Show symbol	Select whether the alarm symbols will be shown in the alarm line. See the chapter *Alarm handling*.
Ack possible	Select whether it will be possible to acknowledge the alarm.
Alarm group	States which alarm group will be shown on the alarm line. The object will be shown in the color specified for the alarm group.
Security level	Select security level for the object.
Frame	Select if a frame shall be drawn around the object.
BG	Select background color for the object.

Tab. 5-22: *Parameter settings for alarm banners*

NOTE The foreground color of the alarm text is defined in the alarm group definition.

Digital Clock

Time object for presenting a digital clock.

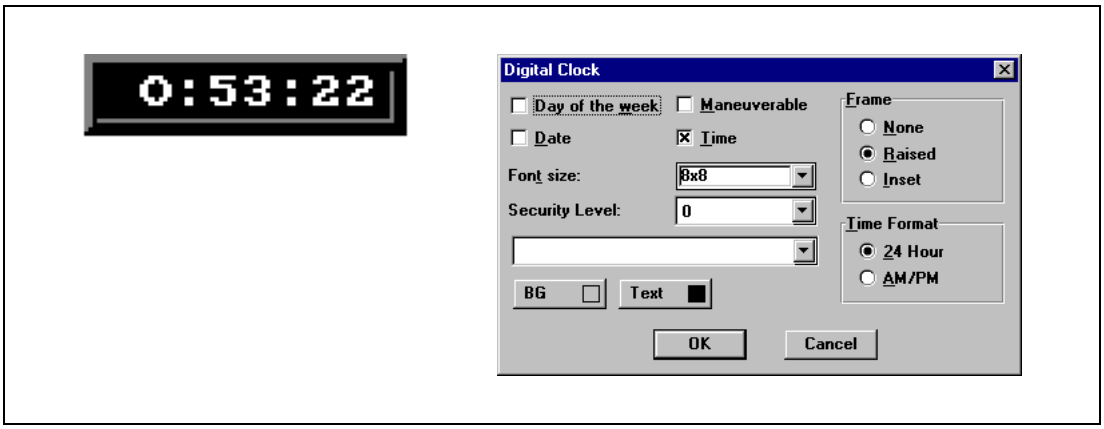


Fig. 5-28: *Dialog box digital clock*

Parameter	Description
Day of the week	Select whether the day of the week will be shown.
Date	Select whether the date will be shown.
Time	Select whether the time will be shown.
Time format	Time can be shown either as AM/PM or a 24-hour clock.
Maneuverable	Select whether the clock can be reset in run-time mode.
Frame	Select if a frame shall be drawn around the object.
Font size	Select font size for the text in the object.
Security level	Select security level for the object.
BG	Select background color.
Text	Select the color of the text in the object.

Tab. 5-23: *Dialog box digital clock*

NOTE A maneuverable Date/Time object must be defined to set the clock.

Analog clock

Time object for presenting an analog clock.

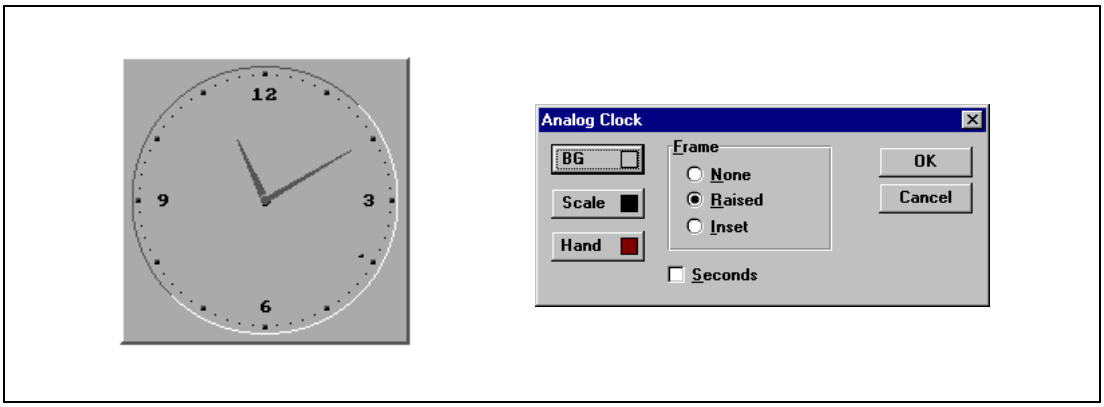


Fig. 5-29: *Dialog box analog clock*

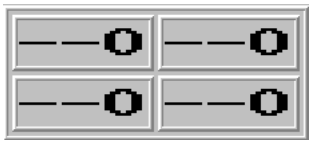
Parameter	Description
Seconds	Select whether the clock will be shown with a second hand.
Frame	Select whether the clock shall be shown with a frame.
BG	Select background color for the object.
Scale	Select the color of the scale in the object.
Hand	Select the color of the second hand in the object.

Tab. 5-24: *Parameter settings for analog clock*

NOTE A maneuverable Date/Time object must be defined to set the clock.

Numeric table

Object for creating a table of analog numeric objects.



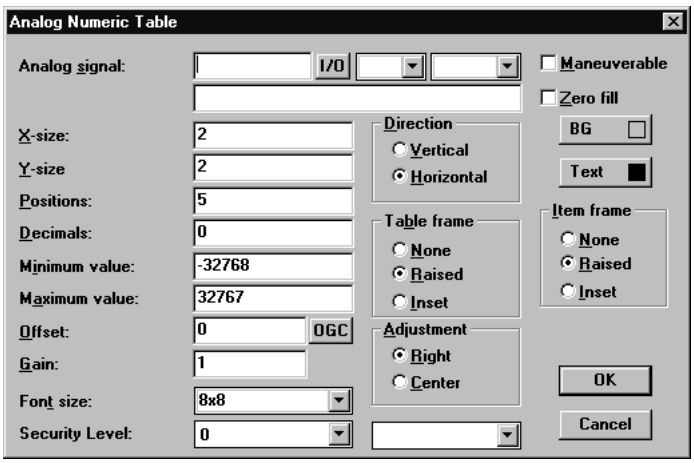


Fig. 5-30: *Dialog box for numeric table*

Parameter	Description
Analog signal	The first signal to be presented in the table.
X-size	State the number of signals to be presented horizontally.
Y-size	State the number of signals to be presented vertically.
Positions	The number of positions you should enter the value with.
Decimals	The number of decimals you should enter the value with.
Minimum value	The minimum limit for maneuvering.
Maximum value	The maximum limit for maneuvering.
Offset and **Gain**	Used to scale register value. See the chapter *General*.
Font size	Select the size of the text in the object.
Security level	Select the security level for the object.
Direction	Select direction for the object, horizontal or vertical. The signals in the table are calculated after this direction.
Zero fill	Select if zeros are to be printed out in empty positions.
Adjustment	Select whether the entry field shall be left aligned or centred.
Maneuverable	Select whether the object can be altered in the run-time mode.
Frame	Select whether the clock shall be shown with a frame.
BG	Select background color for the object.
Text	Select the color of the text in the object.

Tab. 5-25: *Parameter settings for numeric table*

Message

Message object presenting texts from message libraries.

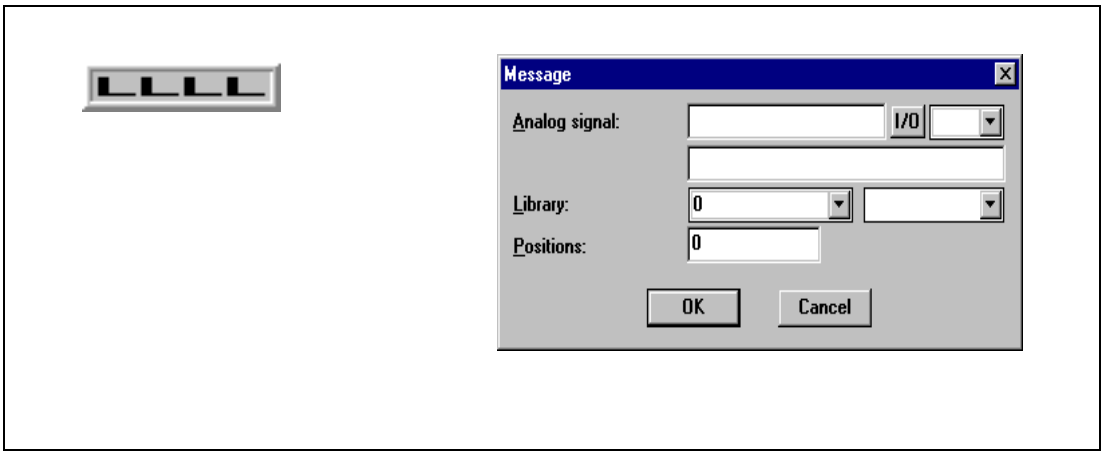


Fig. 5-31: *Dialog box message*

Parameter	Description
Analog signal	Analog signal controlling which text from chosen message library to be shown.
Library table	Select message library. Defined under Message Library in the View menu.
Positions	Number of positions to be displayed.
Adjustment	Select if the text shall be left aligned or centred.
Frame	Select if the text shall be displayed with a frame.
Font size	Select font size of the text.

Tab. 5-26: *Parameter settings for messages*

For more detailed information, see chapter 14, *Message library*.

6 Text-based presentation and maneuvering

Text-based presentation and maneuvering is suitable for generating report print-outs of various kinds. They can be daily reports, status reports etc. For an example of what a report looks like see chapter 9, *Report Print-outs*. Text objects in tabular form are shown in this section. Every object is then described separately. General parameters for the objects are given in chapter 3, *General*.

6.1 Text object

Dynamic text object

Symbol	Object	Description
	Analog	Presents values in numerical form.
	Date/time	Setting of date and time.
	Digital	Switch between two texts depending on the state of a digital signal.
	Multiple-choice	Links a analog signal which can assume eight different states. A text of 30 characters can be linked to each state.
	Jump	Jump to another block.
	Bar graph	Presents values as a bar graph.
	Text	Handles ASCII strings.
	Message	Object controlling which text in a message library to be shown.

Tab. 6-1: *Symbols*

Analog object

Object which presents values in numerical form as whole or decimal numbers.

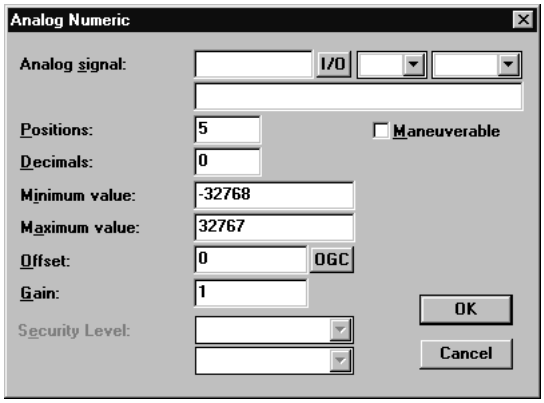


Fig. 6-1: *Dialog box analog object*

Parameter	Description
Analog signal	The signal´s address.
Positions	The number of positions used by the object. Characters such as minus sign or decimal points must also be counted.
Decimals	The number of decimal places which represent the object.
Maneuvrable	States if the object can be altered in run-time.
Minimum value	The minimum value you can enter for an object.
Maximum value	The maximum value you can enter for an object.
Offset and **Gain**	Used to change scale of the register value. See the chapter *General*.
Security level	Select security level for the object.

Tab. 6-2: *Parameters for analog objects*

NOTE Minimum value and maximum value is only valid when maneuvring the object.

Date/time object

Object for setting date and time. Date and time objects collect data from the real-time clocks in the terminal or the CPU.

Fig. 6-2:
Dialog box date/time

Parameter	Description
Day of the week	States if the day of the week will be shown. Requires two positions.
Date	States if the date will be shown. Use the form given in Date/time format in configuration.
Time	States if the time will be shown. Use the form given in Date/time format in configuration.
Maneuvrable	Select if the object can be altered in run-time mode.
Security level	Select security level for the object.

Tab. 6-3: *Parameters for digital objects*

NOTE A maneuverable Date/Time object must be defined to set the clock.

Digital object

Object that can assume two states, zero or one.

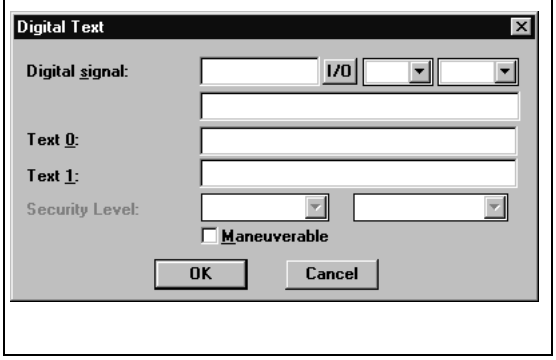


Fig. 6-3:
Dialogbox digital object

Parameter	Description
Digital signal	The signal´s address. See the chapter *General*.
Text 0	The text shown when the signal is zero.
Text 1	The text shown when the signal is one.
Maneuvrable	Select if the object can be altered in run-time mode.
Security level	Select security level for the object.

Tab. 6-4: *Parameters for digital objects*

Multiple choice object

The object is linked to up to three digital signals which together can assume eight different states. A text of seven characters can be linked to each state.

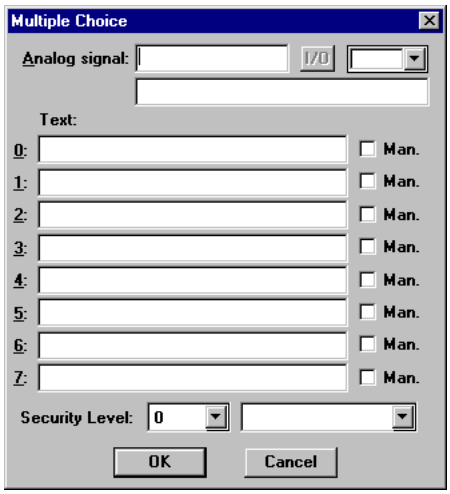


Fig. 6-4:
Dialog box multiple choice object

Parameter	Description
Analog signal	The register controlling which text to be shown.
Text 0-Text 7	The texts that will be shown for each state of the object.
Man 0-Man 7	States whether it is possible to maneuvre to this status in run mode.
Security level	Select security level for the object.

Tab. 6-5: *Parameters for multiple choice objects*

Jump object

Objects for jumping to another block. Hierarchical menus can be built up in the application with jump objects.

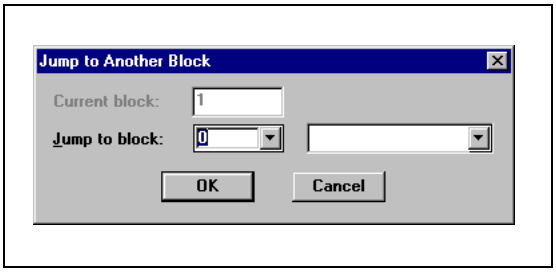


Fig. 6-5: *Dialog box jump object*

Parameter	Description
Current block	States the number of the block left.
Jump to block	States the number of the block being jumped to. Jumping back to the previous block is possible by linking the RET function to a function key (nine levels back). See *Function Key* section.

Tab. 6-6: *Parameters for jump objects*

NOTE If a jump is generated to a block that does not exist in run mode an error message is shown.

Bar graph object

Presents values in the form of a bar graph. The bar will be horizontal so that the maximum number of positions are marked when the value has reached its maximum level.

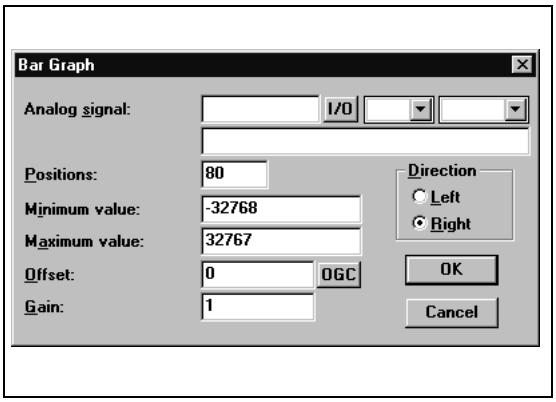


Fig. 6-6: *Dialog box bar graph object*

Parameter	Description
Analog signal	The signal´s address. See the chapter *General*.
Positions	The number of positions which the bar uses.
Direction	States whether the bar will be presented to the right or left.
Minimum value	The minimum value for the object.
Maximum value	The maximum value for the object.
Offset and **Gain**	Used to change scale of the register value. See the chapter *General*.

Tab. 6-7: *Parameters for bar graph objects*

Text object

Object for handling ASCII strings. It is possible to present the text stored in the CPU's data register. Texts should be in ASCII format. The texts can be altered in run-time mode from the terminal by opening and closing the entry field with [ENTER].

The text is converted from 8-bit IBM extended ASCII to Swedish 7-bits ASCII if SW is stated on the command line in System Signals.

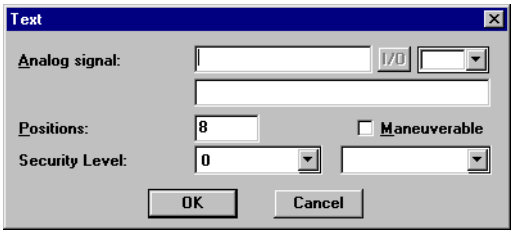


Fig. 6-7:
Dialog box text object

Parameter	Description
Analog signal	The register where text for the first position is stored.
Positions	Each register contains two characters (occupying two positions). The basic setting is 8 positions which is equivalent to four registers.
Maneuvrable	Select if the object can be forcibly set in the run-time mode.
Security level	Select security level for the object.

Tab. 6-8: *Parameters for text objects*

Message

Message object presenting texts from message libraries.

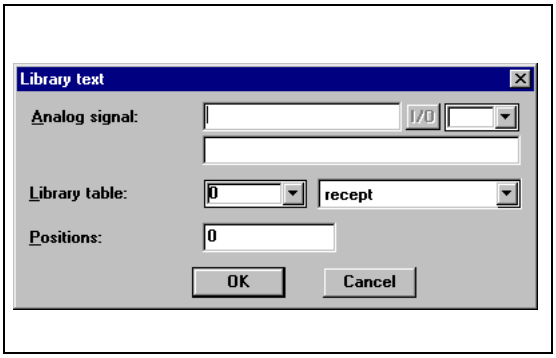


Fig. 6-8:
Dialog box message library

Parameter	Description
Analog signal	Analog signal controlling which text from choosen message library to be shown.
Library table	Select message library. Defined under Message Library in the View menu.
Positions	Number of positions to be displayed.

Tab. 6-9: *Parameters for messages*

For more detailed information, see chapter 14, *Message library.*

7 Alarm handling

This section describes alarm management. The function is used to call the operator´s attention on events in the process that need immediate caution. This function is divided into alarm groups, alarm messages and an alarm list. The alarms can be divided into groups, for example to create an order of priorities. The alarm messages are definitions of the signals that will activate an alarm as well as the alarm message for each signal. The alarm list contains alarms caused in the run-time mode.

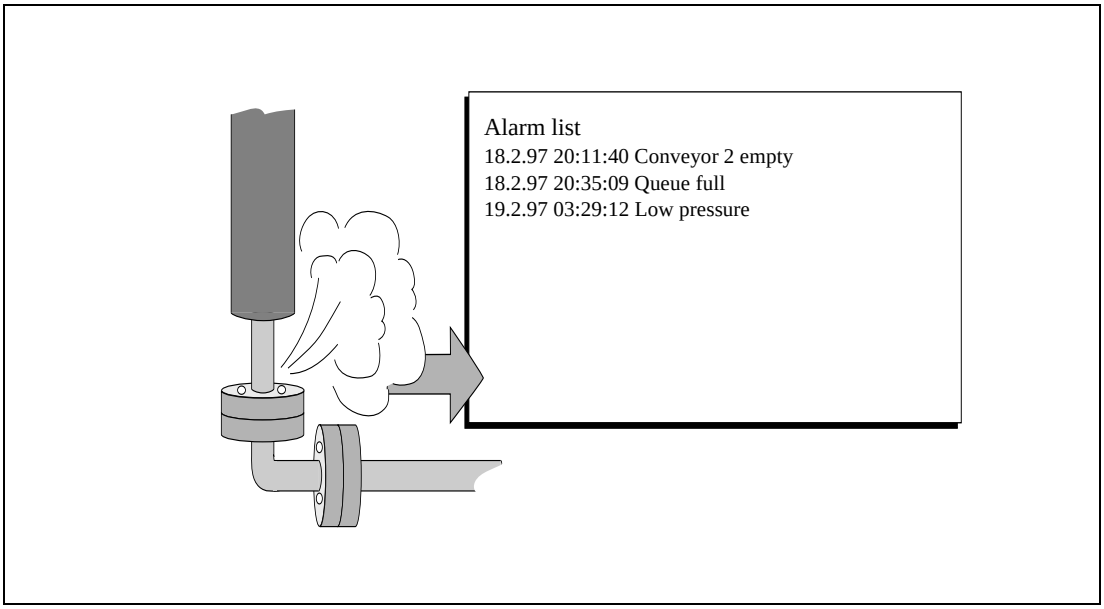


Fig. 7-1: *Alarm list*

7.1 Alarm grouping

Alarms in E900 can be divided into 16 groups. Each group can be allocated a certain color. Alarms can be sorted into groups in the alarm block. It is not necessary to define the alarm groups.

7.1.1 Defining alarm groups

Alarm groups are defined under **Alarm group** in the **View** menu. The attribute for alarms in this group are decided in the following dialog box.

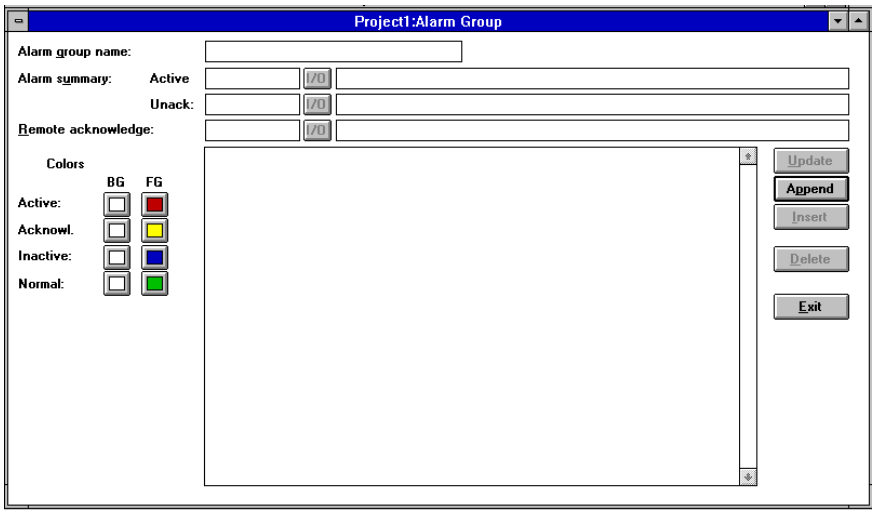


Fig. 7-2: *Dialog box Alarm Group*

Parameter	Description
Alarm group name	Any name for the alarm group.
Alarm summary	Active - Digital signal which is set when there is an active alarm in the group. Unack - Digital signal which is set when there is an unacknowledged alarm in the group.
Remote acknowledge	Digital signal which acknowledges all the alarms in the group simultaneously.
Colors	States the colors for active, acknowledged, inactive alarms as well as the

Tab. 7-1: *Functions in the dialog box Alarm Group*

7.2 Alarm messages

Alarm messages are defined under **Alarms** in the **View** menu. The alarm message is entered here with a maximum of 60 characters. The message can contain dynamic data (just like a text-block) which can be digital or analog. You can define 100–300 alarms depending on the application.

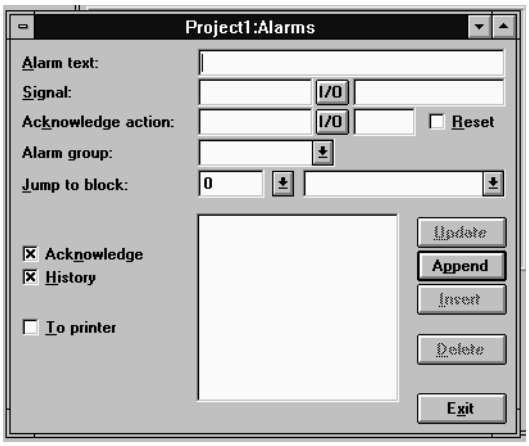


Fig. 7-3: *Dialog box Alarms*

Parameter	Description
Alarm text	An alarm text. Maximum 60 characters.
Signal	The digital signal which when activated generates the alarm is stated here.
Acknowledge action	Digital signal affected when the alarm is acknowledged. Normally the signal is set.
Reset	The above signal is reset when the alarm is acknowledged if Reset is marked.
Alarm group	States the alarm group for the definition.
Acknowledge	States if the alarm should be acknowledged or not. A cross in the square means that the alarm must be acknowledged and no cross means that the alarm functions as an indicator alarm, that is as a form of information.
History	States when the alarm shall be taken out of the alarm list. A cross means that the alarm remains in the list until the alarm list is full. No cross means that the the alarm is taken out of the list when it is acknowledged and is no longer active. If the **Acknowledge** parameter is not marked then the alarm is taken out of the list as soon as it is not active.
To printer	Whether the alarm message should be printed out directly if the alarm status changes is stated here.
Block jump	State a block number of a text or graphic block. This makes it possible to give the operator a “help page” containing information about the alarm and possible actions. 0 means that no block are linked to the alarm. See the chapter *Run-time mode*.

Tab. 7-2: *Parameters for the alarms*

7.3 Configuring the alarm list

General settings for the alarm list are made in **Alarm settings** in the **Setup** menu. Alarms take different amounts of space in the alarm list depending on the length of the alarm text and the number of objects in the alarm text. Each object in the alarm list takes 19 bytes + the number of characters in the alarm text.

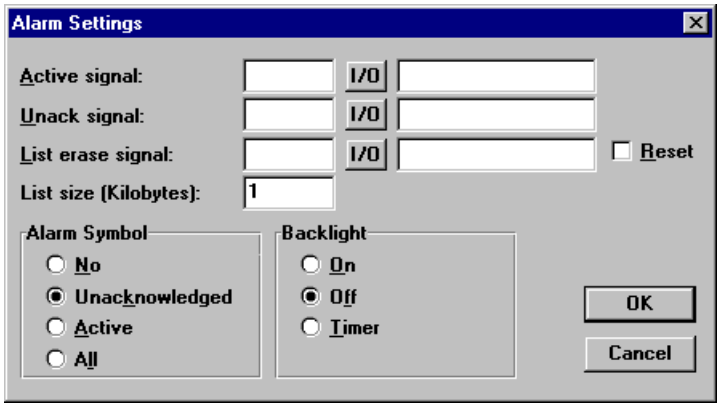


Fig. 7-4: *Dialog box Alarm Settings*

Parameter	Description
Active signal	States the digital signal which the terminal sets when there is an active alarm.
Unack signal	States the digital signal which the terminal sets when there is an unacknowledged alarm.
List erase	States the digital signal which when set erases the whole alarm list.
Reset	Means that the signal stated on List erase is reset when the alarm list is erased.
List size	States the list size in number of kbytes.
Alarm symbol	States when the alarm symbol should be shown, see table below. ■ALARM■ is shown in the Textblock and a red bell, , in the top right-hand corner of the display is shown in the Graphicblock.
Backlight action	State when the background light shall be turned on at alarm. **On** means that the backlight is turned on when the alarm symbol is shown. Default setting. **Off** means that the backlight is not affected by alarms. **Timer** means that the backlight is on when a new alarm is activated and turns of when the time for the screen save is out.

Tab. 7-3: *Functions in the dialog box Alarm Settings*

Option	Alarm symbol shows when	
	active	unacknowledge
NO		
UNACK		•
ACTIVE	•	
ALL	•	•

Tab. 7-4: *Options of the function Alarm Symbol*

8 Historical trends

This section describes historical trends. This function means that analog values from the PLC system are collected and presented in a trend object while running. The presentation is in the form of a curve. The values collected are stored in the terminal's project memory.

It is possible to define several independent trend curves in the same or different blocks. The total number is limited by the project memory's capacity and the extent to which it is used.

Such things as the time interval between collection of data and how many values are to be stored are defined in the trend object.

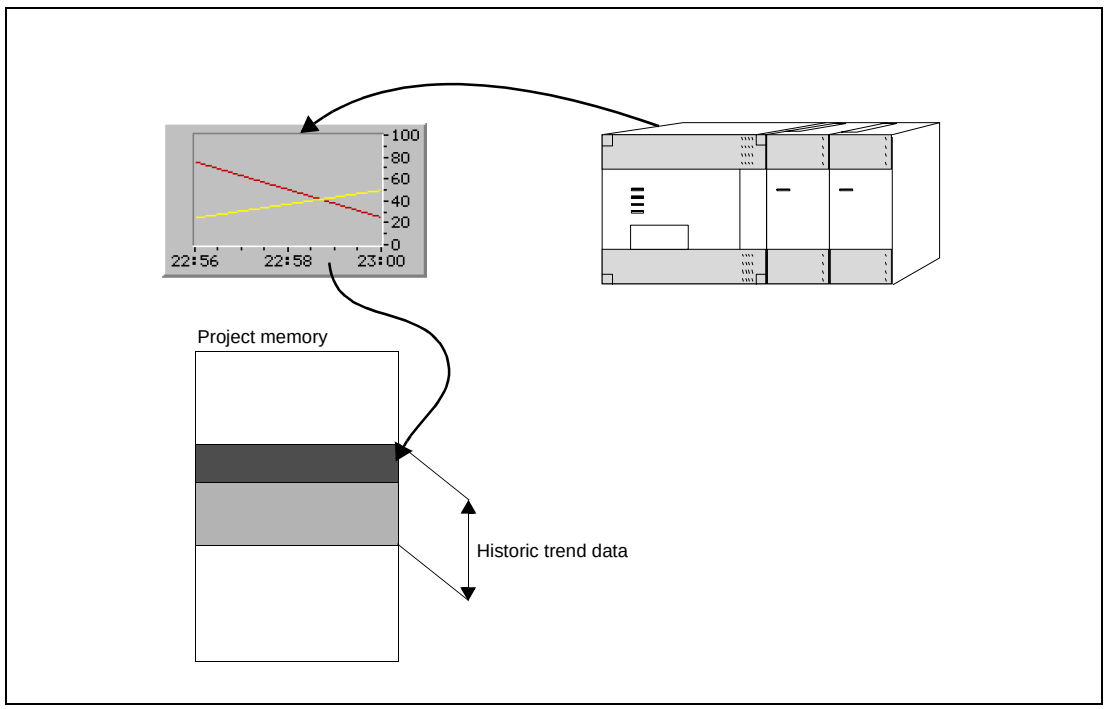


Fig. 8-1: *Trend diagram*

Calculate the trend size

If you want to know how much the trend data will occupy of the project memory the following calculation must be made.

- S = TOS + AK (28 + (645 * ((AS / 100) + 1))
  - TOS=TrendObjektSize, if all parameters for a trend object are changed the TOS will be 320 bytes.
  - AK=The number of defined curves in the trend object.
  - AS=The number of Samples is rounded up to the nearest hundred.
  - S=The number of bytes.

8.1 Defining trend objects

Trend objects are defined as other dynamic objects in a block. The object can be linked to one to six analog signals. Unlike other objects, trend objects must be given a name with 1 to 8 characters. The first character must be a letter or a number. The standards for naming files in MS DOS are applied. The following parameters are defined for a trend object:

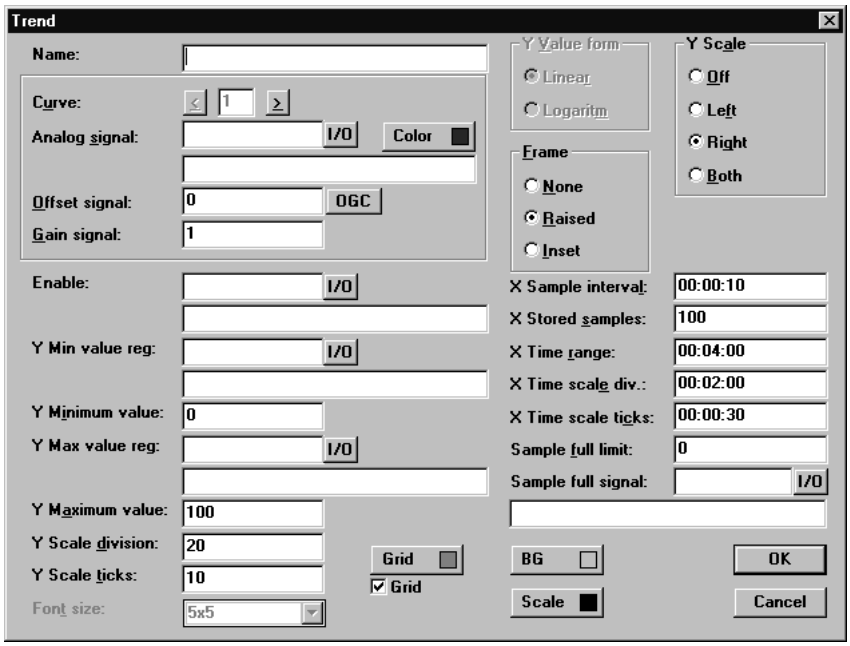


Fig. 8-2: *Dialog box trend object*

Parameter	Description
Name	State the name of the object. The name must be unique for each object.
Curve	Select the curve you want to edit (1-6).
Analog signal	The analog signals the object shall sample and present the values for. Only 16-bit numbers can be used.
Color	Select a color for each curve.
Offset and Gain	Used to scale register value. See the chapter *General*.
Enable	The digital signal which when set starts the collection of data. Data collection ceases if the signal is set to zero. The parameter needs not be stated.
Y Min value reg.	The minimum value on the Y-axis loaded from the stated PLC register.
Y Minimum value	The minimum value on the Y-axis. (Ignored if Y Min value reg. is defined).
Y Max value reg.	The maximum value on the Y-axis loaded from the stated PLC register.
Y Maximum value	The maximum value on the Y-axis. (Ignored if Y Max value reg. is defined.)
Y Scale	States which side of the Y-axis shall be shown, to the left, right, both or neither.
Y Scale division	The interval between the scale number on the Y-axis.
Y Scale ticks	The interval between the scale ticks displayed.
X Sample interval	States the time interval between the collection of data. Minimum value is 1 second.
X Stored samples	States how many values shall be stored. The maximum number of values are 4800.
X Time range	States the time interval to be shown in the trend diagram.
X Time scale division	The interval between the scale number on the X-axis.
X Time scale ticks	The interval between the scale ticks displayed.
Sample full limit	State the number of samples when the samples full signal shall be set.
Sample full signal	Select a signal which will be set when the number of samples reach the value stated in the parameter Sample full limit.
Frame	Select if a frame shall be prited around the object.
BG	Select background color for the object.
Scale	Select the color of the scale in the object.
Grid	Select if grid shall be displayed in the object.
Grid color	Select a suitable color for the grid.

Tab. 8-1: *Parameter settings for trend objects*

NOTE

If you copy a block containing a trend object you must change the name of the trend object. Two trend objects may not have the same name.

Presentation in run-time mode

Place the cursor on the trend object and press [ENTER] to activate the presentation of historic data. For more information see chapter 18, *Run-time Mode*.

Transferring trend data

This function is used on a personal computer with the Windows 3.1 and Windows 95/NT. Trend data and recipes can be transferred to and from a personal computer with the program *Transfer* (an icon in the program group HMI Tools) for calculating statistics and other types of presentation and storage.

Trend files and production methods can for example be opened directly in Excel for statistical calculations. Values can also be changed and thereafter transferred by copying or saving files in the unit. See the manual for *MAC Programmer plus (MP+)*.

Trend files

The name of the trend files is given for each trend when the trend objects are defined. The file will be allocated the extension .SKV. The format for each line in the trend file is as follows; DDDD;TTTT;AAAA;BBBB;CCCC;DDDD;EEEE;FFFF:

Format	Description
DDDD	Date format given in Setup.
TTTT	Time format given in Setup.
AAAA	Trend curve 1.
BBBB	Trend curve 2 (if defined).
CCCC	Trend curve 3 (if defined).
DDDD	Trend curve 4 (if defined).
EEEE	Trend curve 5 (if defined).
FFFF	Trend curve 6 (if defined).

Tab. 8-2: Format for each line in the trend file

The oldest value is shown on the first line of the file and the latest on the last line. The .SKV format can be imported directly to Microsoft Excel. You can use the diagram manager to create statistical diagrams in Excel. It is not possible to alter something in the file and then send it back to the terminal. The picture below shows an example of how it can appear in Excel.

TR000_02.SKV

	A	B	C	D	E	F	G	H	I
1	98-01-28	07.39.00	0	0	0	0	0	0	
2	98-01-28	07.39.30	40	55	15	30	45	60	
3	98-01-28	07.58.20	40	55	15	30	45	60	
4	98-01-28	08.04.00	40	55	15	30	45	60	
5	98-01-28	08.06.20	40	55	15	30	45	60	
6	98-01-28	08.09.35	40	55	15	30	45	60	
7	98-01-28	08.15.55	55	40	80	65	50	35	
8	98-01-28	08.37.25	55	40	80	65	50	35	
9	98-01-28	08.38.30	55	40	80	65	50	35	
10	98-01-28	08.45.30	55	40	80	65	50	35	
11	98-01-28	08.46.15	40	20	10	50	20	30	
12	98-01-28	08.49.15	40	20	10	50	20	30	
13	98-01-28	08.49.50	10	20	30	40	50	60	
14	98-01-28	08.51.50	10	50	30	40	50	60	
15	98-01-28	09.01.10	50	50	40	30	20	10	
16	98-01-28	09.01.20	60	50	40	30	20	10	
17	98-01-28	09.05.30	60	50	40	30	20	10	
18	98-01-28	09.06.45	60	50	40	30	20	10	

Fig. 8-3: Microsoft Excel

9 Report print-outs

This section describes how to make a report print-out from the terminal. For production follow-up it is simple to create different types of reports such as daily reports and event reports. The illustration below shows the principle for generating a daily report.

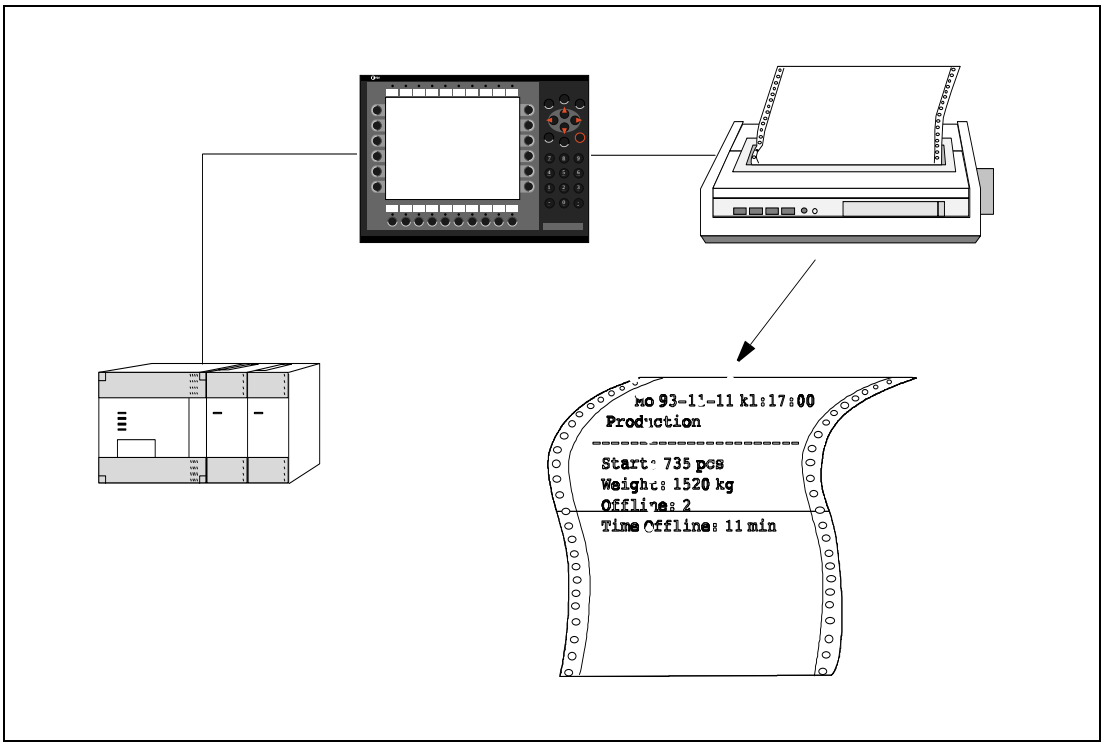


Fig. 9-1: *Create report printouts*

9.1 Create report print-outs

The reports are created as Textblocks with static and dynamic text. The report has a maximum width of 150 characters. In the textblock you enter a free text, e.g. table head and other static text which always will be printed. To get dynamic values from the process in the report you define dynamic objects, presenting the value for the signal the object is linked to. Se the chapter 6, *Text-based presentation and maneuvring*. Via time channels it is possible to decide when the report is to be printed.

9.2 Defining the print-out

The **Printer signal** parameter in the block header gives the signal which when set activates the print-out. A **Completion signal** is also given here which is set when the print-out is complete. This signal is instead set to zero if **Reset** is marked.

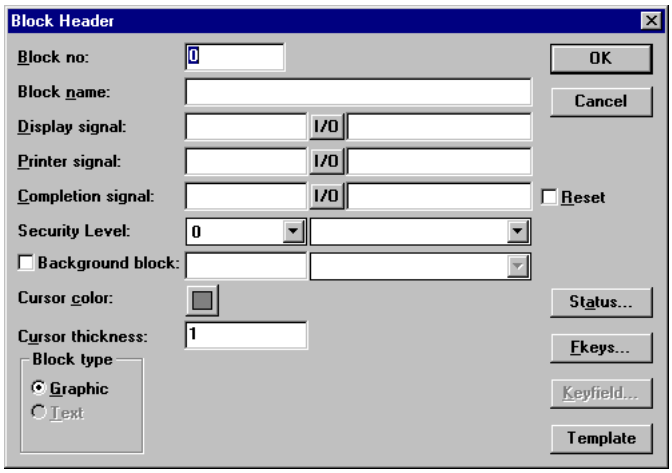


Fig. 9-2: *Dialog box block header*

NOTE Graphic blocks can only be printed when it´s displayed (screen dump).

9.3 Printer settings

The settings for the printer must be made under **Printer settings** in the **Setup** menu. For more detailed information concerning the printer refer to the manual for the printer used. The printer must support IBM extended ASCII character set. If graphic blocks are printed the printer must support Epson FX-80 graphics.

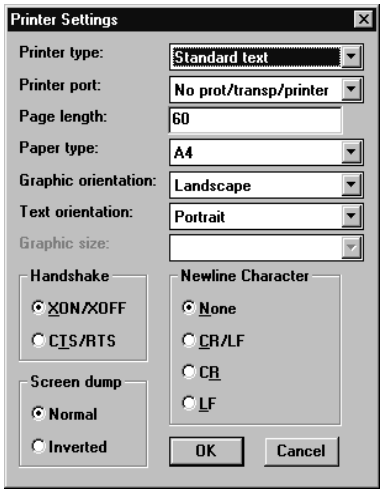


Fig. 9-3:
Dialog box printer settings

Parameter	Description
Printer type	Select printer, none or installed printer, e.g. HP PLC5.
Printer port	Select the communication port connected to the printer.
Page length	The number of lines to be printed out before form feed. Form feed never takes place if the page length is set to 0.
Paper type	Select paper type according to installed printer.
Graphic orientation	States if the graphic printout should be portrait or landscape if a PCL5 compatible printer is used.
Text orientation	States if the report printout should be portrait or landscape if a PCL5 compatible printer is used.
New line character	End of line character; CR/LF, CR, LF or none.
Handshake	States whether handshaking between the printer and terminal will take place with XON/ XOFF or CTS/RTS.
Screen dump	Possibility to printout the screen, normal or inverted.

Tab. 9-1: *Parameter settings for printers*

9.4 Setting a communication port

Settings for the communication port must be made under Port parameters, Print/Transp/No prot in the Setup menu. See the manual for the printer concerned for the correct setting.

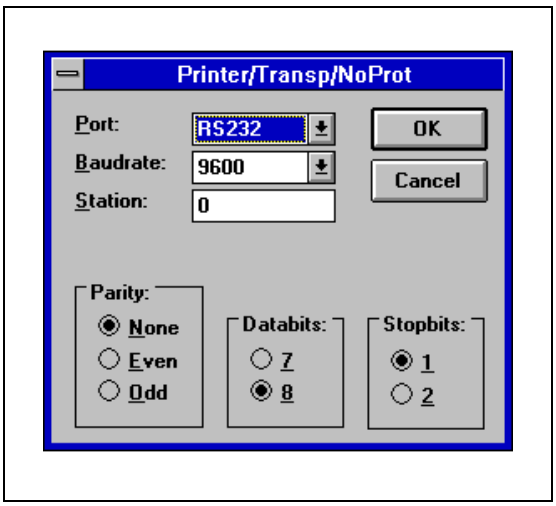


Fig. 9-4:
Dialog box communication port

Parameter	Description
Port	Select the communication port, RS232 or RS422.
Baudrate	Select the transfer rate for communication. Must be the same as the external unit.
Station	State the number for the terminal.
Parity	Select parity. Must be the same as the external unit.
Databits	Select number of data bits. Must be the same as the external unit.
Stopbits	Select number of stop bits. Must be the same as the external unit.

Tab. 9-2: *Parameter settings for communication ports*

10 Recipes

This chapter describes the function recipe handling. The recipes function means that all dynamic data in a block, that are signals and their values, can be saved to the file in the run-time mode. The operator can then down-load the file to the PLC system. The PLC system then starts to work with the loaded values. By using the recipe handling function a large amount of parameter settings can easily be reused. The user creates a recipe library containing files with different parameter settings. This function makes time-critical production more efficient. For example when you often shift between similar products e.g. when producing a unit in different colors. The recipe files can be created from the terminal or from the PLC system or from a PC with the software HMI Tools.

Recipe files are stored in the terminal. To use recipes, the functions for storing, loading and erasing recipes must be linked to function keys, see the chapter 12, *Function keys.*

The diagram below shows the principle for managing recipes.

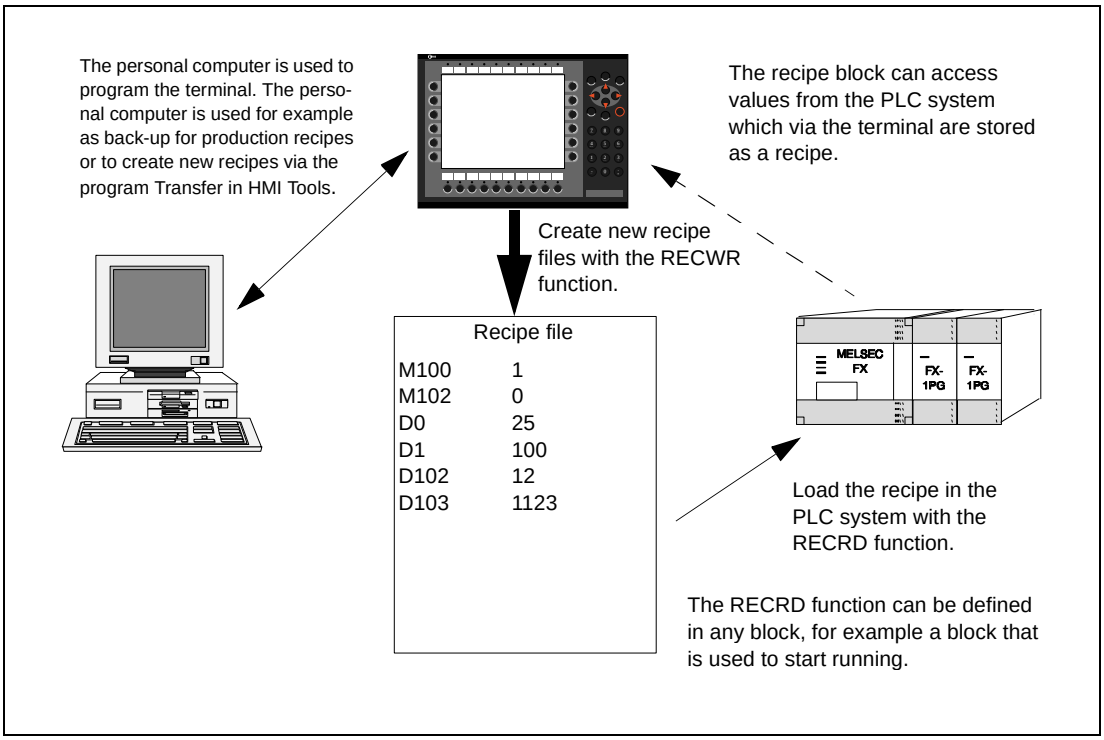


Fig. 10-1: *Recipe handling*

Calculate the recipe size

Use the following equation to calculate how much the recipe will occupy the project memory.

$S = 80 + \Sigma_n (2 * IOG_i + 18)$

S = number of bytes. If S calculated will be less than 360, S will be set to 360.
n = the number of IO-series
IOG_i = the number of IO in each IO-series
Σ = the sum of the n-different IO-series

Example ▽

Our recipe has the following series D0–D109, D200–D499 and M0–M99.
The calculation will be:
Calculate the sums of the different series:
1074 = (2 * 110 + 18) + (2 * 300 + 18) + (2 * 100 + 18)
Then calculate the sum:
S = 80+ 1074
S = 1154 bytes

△

10.1 Creating a recipe

Which block or blocks are to be used to save the recipe is decided when the application is programmed. All those signals that are to be included in the recipe are defined in the block that will be saved as a recipe. All dynamic values in the block are stored in the recipe file. All digital and all analog objects except trend objects can be used as recipe parameters.

A jump is made in run-time mode to the block which contains the recipe parameters. Enter the required values in the dynamic object and press the function key which is linked to RECWR. The name of the file is given in the entry field shown. The recipe file is stored in the terminal.

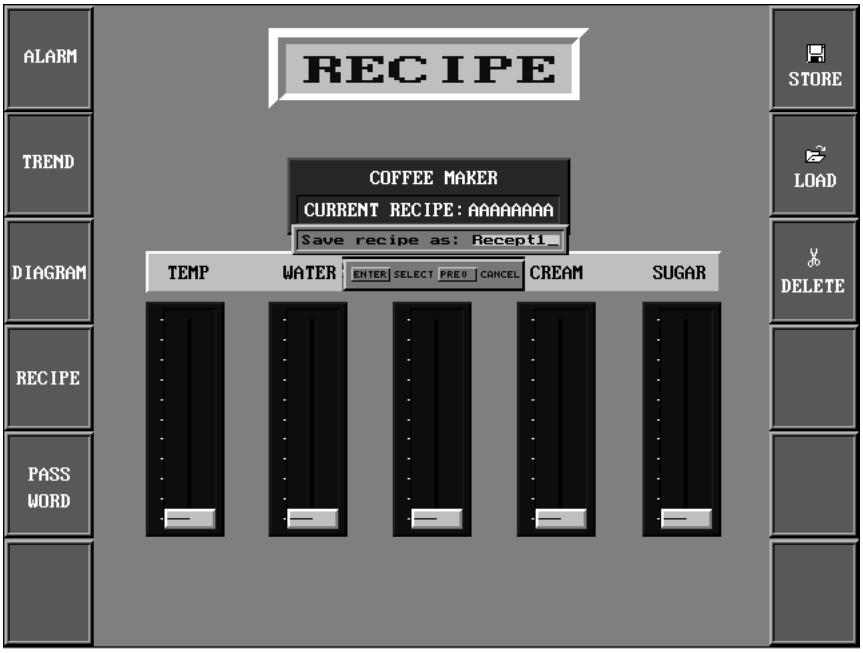


Fig. 10-2: *Storing a recipe*

10.2 Transferring a recipe to the PLC system

The recipe is transferred to the PLC system in run-time mode with the RECRD function. This means that the signals and their values stored in the file are transferred to the PLC system. When the function key for RECRD is pressed a list of available recipe files to choose from is shown. Select one and press [ENTER]. The PLC system will then run with the down-loaded values.

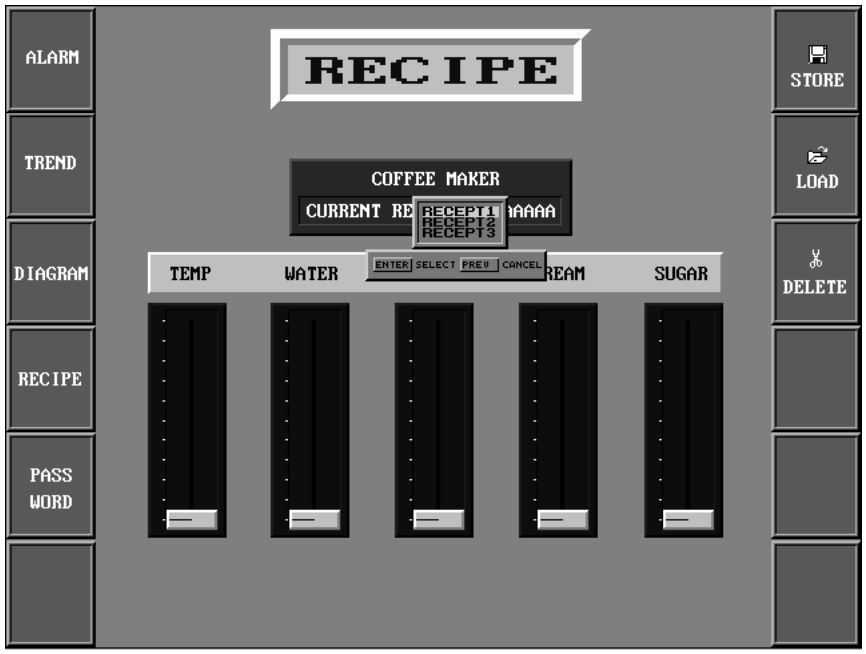


Fig. 10-3: *Transferring a recipe*

10.3 Erasing a recipe

Any named recipe can be erased in run-time mode from the terminal memory with the RECDEL function. Press the function key connected to RECDEL. Available recipe files to choose from are now shown. Select the file to be erased and press [ENTER]. Confirm the erasure by pressing [ENTER] or abort by pressing [PREV].

10.4 Using recipes in a personal computer

The terminal is treated as a unit in the personal computer with the program Transfer (in the program group HMI Tools). This means that the personal computer can be used for making safety copies of files in the terminal for example recipe files. New recipes can be created in the personal computer and be copied to the terminal.

The files are stored in the .SKV format in the personal computer which can be brought up in Excel. Files can be edited in Excel and then used in the plant again.

The file has the format [OPERANT][;][NAME] and it must end with the instruction END. See the example below.

EXAMPLE ▽

```
M100;1
M102;0
D0;25
D50;12
END
```

△

For further information we refer to the manual for *MAC Programmer plus (MP+)*.

10.5 Loading and saving recipes via the PLC program

By using a control block in the PLC system you can load, save and delete recipe files in the terminal. The files created by the PLC program are compatible with the recipe files saved in the terminal. This means that recipe files saved by the PLC program can be loaded in the terminal and vice versa. The control block is described in the following table.

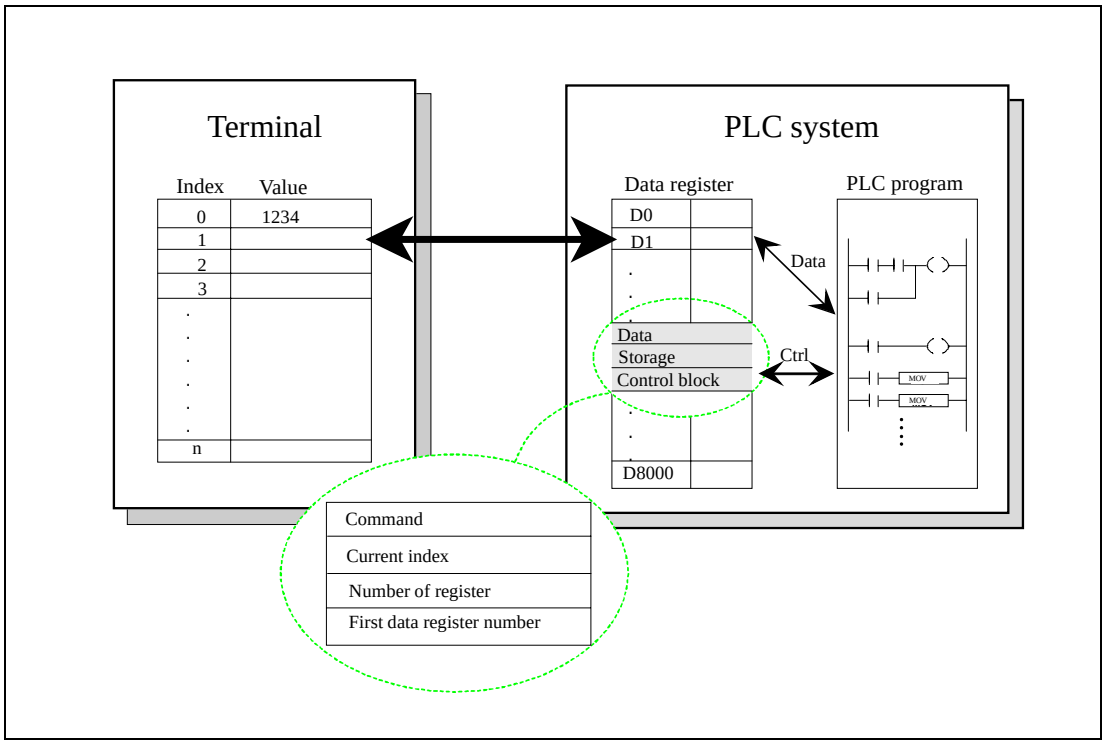


Fig. 10-4: *Control block*

Register	Content	Description
Dn0	Command	Command register set by the PLC Available commands: 0. No command. 1. Save recipe to the terminal. 2. Read recipe from the terminal. 3. Delete recipe from the terminal.
Dn1	Result code	Handshake register set by the terminal. 0. Ready for new command. 1. OK 2. Recipe file write error. 3. Recipe file not found.
Dn2	File name char. 1-2	Recipe file name in the terminal.
Dn3	File name char. 3-4	
Dn4	File name char. 5-6	
Dn5	File name char.7-8	
Dn6	Start data register	The first data register to be read/written to/from recipe file.
Dn7	Number of registers	Number of registers to read/write to/from recipe file.

Tab. 10-1: *Register*

In the **Setup** menu, **System signals** you define the first register in the control register area. This register and the six following registers are used as a control register.

To perform commands, the following handshake procedure must be carried out:

① The handshake register must be 0. If it is not, make sure that the command register is set to 0.

② Write the command in the command register.

③ Wait for completion or error code in the handshake register.

④ Set the command register to 0. After this, the terminal will set the handshake register to 0.

Example ▽ The PLC wants to save D100 to D149 as recipe name “RECIPE1” when M0 is activated. The control block starts at D10. The PLC program should be as follows.

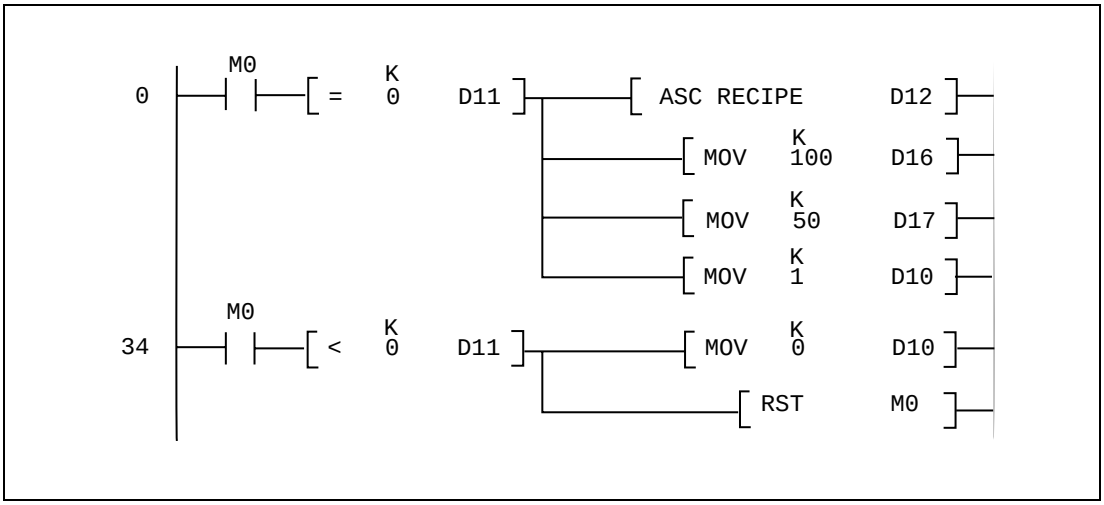


Fig. 10-5: *Example of a program*

△

NOTE Recipe created in the PLC system can include maximum 1000 registers.

11 LEDs

The terminal has 20 LEDs built in, ten are located near the function keys and ten are independent. The LEDs are connected to a register. The content of the register determines the color and any blinking of the LED according to the following table.

Register value [Hex]	Register value [Dec]	Blinking frequency [Hz]	Color
00	0	—	None
01	1	—	Green
02	2	—	Red
11	17	5	Green
12	18	5	Red
21	33	2.5	Green
22	34	2.5	Red
31	49	1.2	Green
32	50	1.2	Red
41	65	0.6	Green
42	66	0.6	Red

Tab. 11-1: *LED-register*

The location of the LEDs is shown in chapter 12, *Function keys*.

11.1 Run-time mode

The LEDs according to the table above depending on the value in the analog register linked to the LED. See chapter 18, *Run-time Mode*.

12 Function keys

This section describes how to use the function keys. A function key is linked to a signal (e.g. M100) or a function by writing in its address after each key. The digital signal linked to a function key is activated according to the function linked to the key when it was defined.

The terminal has 22 function keys (F1 to F22) built in.

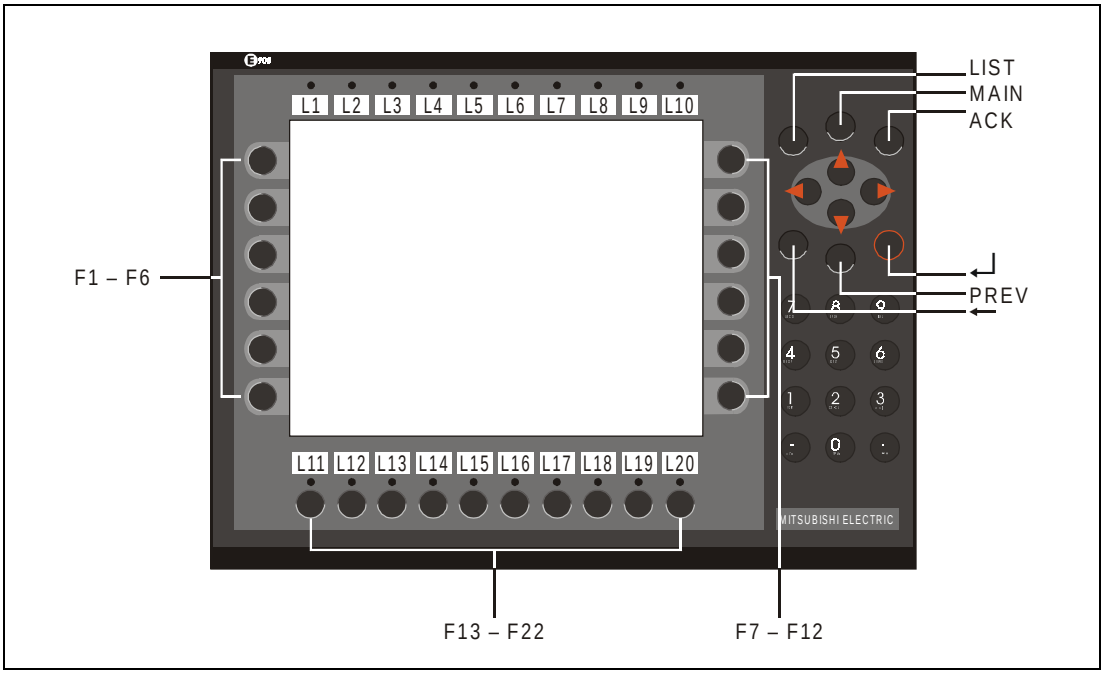


Fig. 12-1: *Function keys*

12.1 Definitions

There are two ways of defining the function keys; *globally* or *locally.* Global function keys are defined and used in the whole application, that is they apply in all the blocks. Local function keys are defined and used in a single block. A global definition is always available in the run-time mode assuming that the block shown on the display does not have a local definition for the function key in question. Local definitions have a higher priority than global ones. Function keys are defined under **Function keys** in the **View** menu.

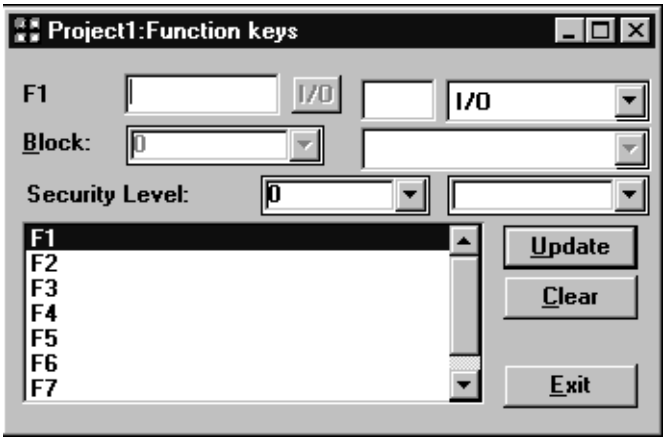


Fig. 12-2: *Definition of function keys*

Parameter	Description
F1–F22	The signal or function to be linked to the function key is stated here. Both global and local function keys can be linked to different functions. The table below shows which functions can be linked to function keys.
Block	The block to which a jump takes place if the function ”BlockJump” is chosen is stated here.
Security Level	Function keys can be defined with a security level. The security level means that the operator must log in with a password for the same or a higher security level to be able to use the function key.

Tab. 12-1: *Parameter settings for function keys*

Function	Description
AlarmAck	Acknowledges alarms in the alarm list.
AlarmInfo	Jump to a block connected to the alarm. See chapter 7, *Alarm handling*.
AlarmList	Shows the alarm list (block 990).
BlockJump	Jumps to the block with number nnn.
Diagnostics	Shows the information page. See chapter 22, *Appendix*.
Decrement	Resets digital objects. Decreases analog object values.
Enter	Makes a jump for a jump object, end entering of analog object, alter status for digital object, entry field text object, option list for multichoice object.
Info	Shows the min. and max. values for analog objects in the textblock in the run-time mode.
Increment	Sets digital objects. Increases analog object values.
IO	With the function **IO event** it is possible to select how the key should affect current signal. The following alternatives are available: **Momentary** means that the signal is set as long as the key is active. **Toggle** means that the signal is set and reset respectively when the key is activated. **On** means that the signal is set when the key is activated and remain set. **Off** means that the signal is reset when the key is activated and remain reset.
Login	Logging in (state security level).
Logout	Logging out.
LogSet	Change password.
Main	Show the main menu (block 0).
Momentary	The digital object the curcor is placed on is active as long as the key is pressed.
PageDown	Function for flipping backwards through the pages (Textblock and alarm page).
PageUp	Function for flipping forwards through the pages (Textblock and alarm page).
RecDel	Erase recipe.
RecFrC	Load recipe from memory card.
RecRd	Create recipe.
RecToC	Store recipe from memory card.
RecWr	Store recipe.
Return	Show the previous block, works for nine levels.
Set	Set entered signal to entered value.
Zoomdn	Reduce the text size in the alarm list.
Zoomup	Increase the text size in the alarm list.

Tab. 12-2: *Function and description of the built-in keys*

12.2 Jump to block with the function keys

This function makes it possible to use the function keys for jumping to blocks without using the Display signal. When the key is defined, locally or globally, you select BlockJump in the list of options. Changing block via function keys is the easiest way to change block. The method do not occupy any digital signal in the PLC system.

13 Time control

Time control is a function for setting and resetting digital signals relative to the real-time clock. This function is used for controlling events in the process at specific times through the terminal. The time channels replace the time relays and 7-day clock.

13.1 Defining the time channels

The time channels are defined under **Time channels** in the **View** menu.

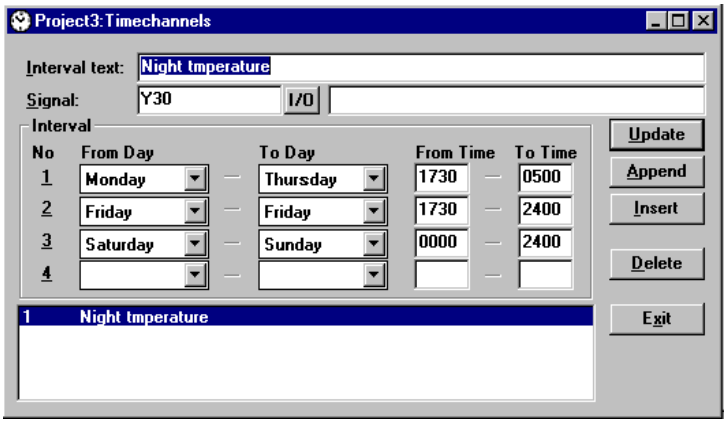


Fig. 13-1: *Dialogbox time channels*

Parameter	Description
Interval name	Enter any text for the time channel.
Signal	A digital signal which is set during the stated time period is stated here.
Interval	The day and time for the interval is stated here. Four intervals can be defined for each time channel.

Tab. 13-1: *Parameter settings time channels*

In the figure above we have defined one time channels. The time channel has the text Night temperature and is linked to the signal Y30. The signal will be set Monday to Thursday 17.30 to 05.00, Friday 17.30 to 24.00 and Saturday to Sunday between 00.00 and 24.00. During these periods the temperature in the building will be decreased.

13.2 Run-time mode

The page for the time channels can be found in system block 991. Activate the block through a digital signal or by jumping there. See chapter 18, *Run-time Mode*. In this block the time channel settings can be changed in run-time mode.

13.3 Real-time clock

The real-time clock in the terminal is configured under **Date/time format** in the **Setup** menu.

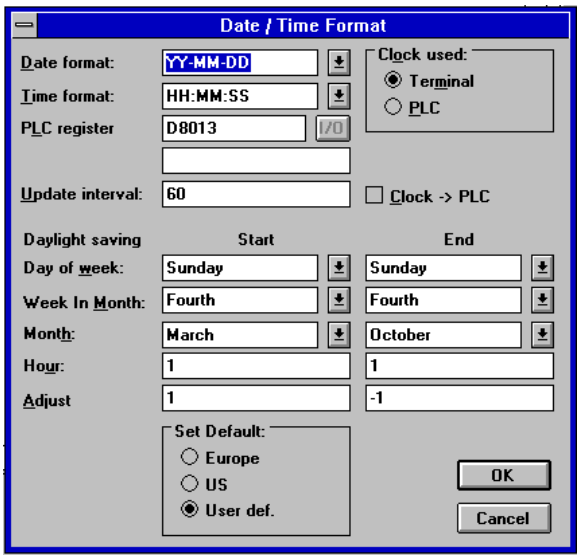


Fig. 13-2: *Dialog box date / time format*

Parameter	Description
Date format	Select the format for the presentation of the date.
Time format	Select the format for the presentation of the time.
PLC register	Start address for storing date/time. Only used if Clock→ PLC is set to YES.
Update interval	The updating interval in seconds.
Daylight savings	Setting for the beginning and end of daylight saving time.
Clock used	Select whether the clock in the terminal or PLC system will be used.
Clock → PLC	State whether the clock in the terminal shall be transferred to the register in the PLC system.

Tab. 13-2: *Functions in the dialogbox date / time format*

NOTES

If you want to change the clock in the PLC system from the terminal, this must be done in run mode.

The terminal automatically adjusts the time for daylight savings- and standard time.

14 Message library

With the function message library you create message tables where values between 0 and 65535 are linked to texts. The function is used for presentation of each step in a sequence control. Another use is presenting error codes. An analog signal generates the error codes which are linked to texts in a message library. The function can also be used for operator communication.

Message library is one or several message tables containing up to 512 text strings each. Each text string can have a length of 40 characters.

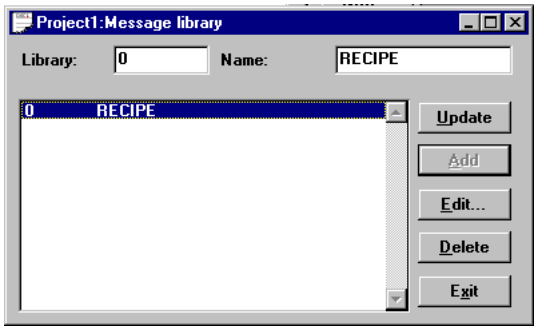


Fig. 14-1:
Dialog box message library

Parameter	Description
Library	State a number of the message library.
Description	State a name of the message library.

Tab. 14-1: *Parameter settings for message libraries*

You edit a message library by marking a library and click on Edit. Several editing windows can be displayed at the same time.

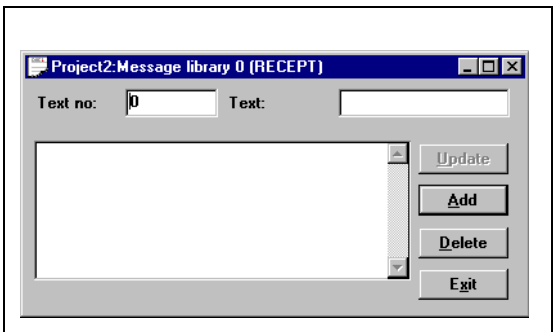


Fig. 14-2:
Editing window

Parameter	Description
Text no	State a number of the text, 0–65535.
Text	Freely defined text which is displayed when current signal includes the text number for the text.

Tab. 14-2: *States in the editing window*

Example ▽

The following example shows how the function may be used. In a sequence control we want to display a text for each step in the sequence.

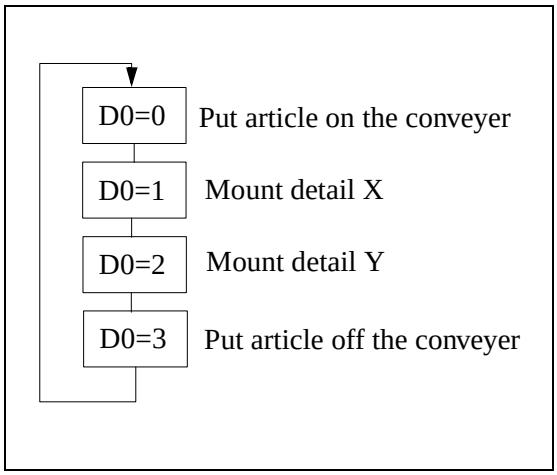


Fig. 14-3:
Sequence and displayed text

Start by creating a message library named Line1.

① Select **Message library** in the **View** menu.

② Define a number, in this case 0, and a name of the library, in this case Line1.

③ Click on Add.

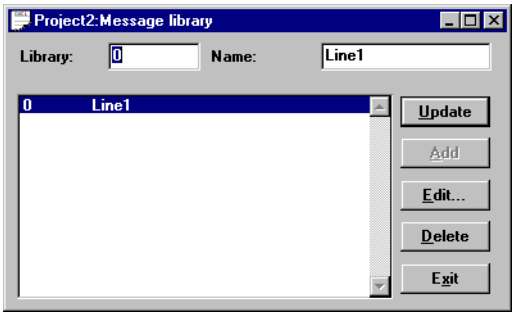


Fig. 14-4:
Adding a message library

A message library named Line1 is now created. Thereafter you define the texts in the library.

④ Mark the library, line1, and click on Edit.

⑤ In this dialog you define text number and text. In the field **Text no** you state a value of the analog signal linked to the message object. In the field **Text** state the text displayed in the message object.

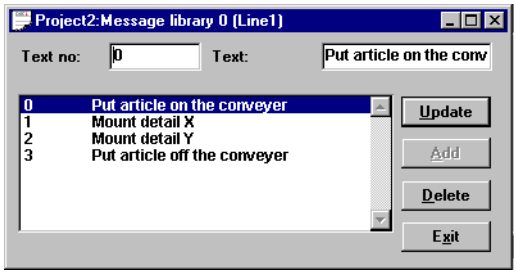


Fig. 14-5:
Text number and text

When the message library is defined, you create a message object in the project. Message objects can be defined in both text and graphic blocks.

⑥ Select the **Message object** in the toolbox. Place the cursor where you want to place the object and click.

⑦ State the analog signal to control which text is to be displayed.

⑧ In the field library you select message library.

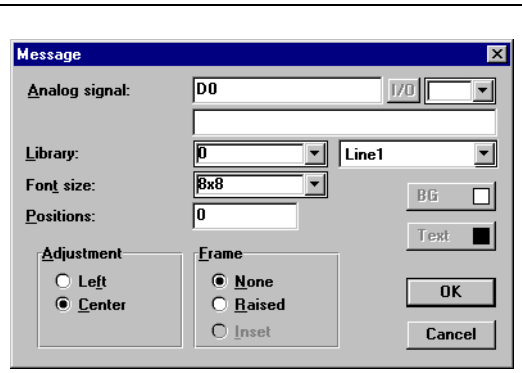


Fig. 14-6:
Link to an analog signal

△

15 Communication

This section describes the various communication configurations in which the terminal can be used. The terminal has 2 series channels either of which can be programmed.

15.1 The terminal in the transparent mode

The programming/printer port on the terminal is used in the transparent mode to connect a unit working in parallel with the PLC system. This unit can be another terminal or a personal computer with for example the MELSEC MEDOC programming tool or a superior operating system.

The transparent mode works together with the FX-CPU port, A-CPU port, A-series communication ports with C24 protocol. For more information see the C24 manual concerned.

15.1.1 Connecting a personal computer or other computer system

Personal computers with the MELSEC MEDOC programming tool or other computer system are connected directly to the terminal's programming/printer port by the programming cable.

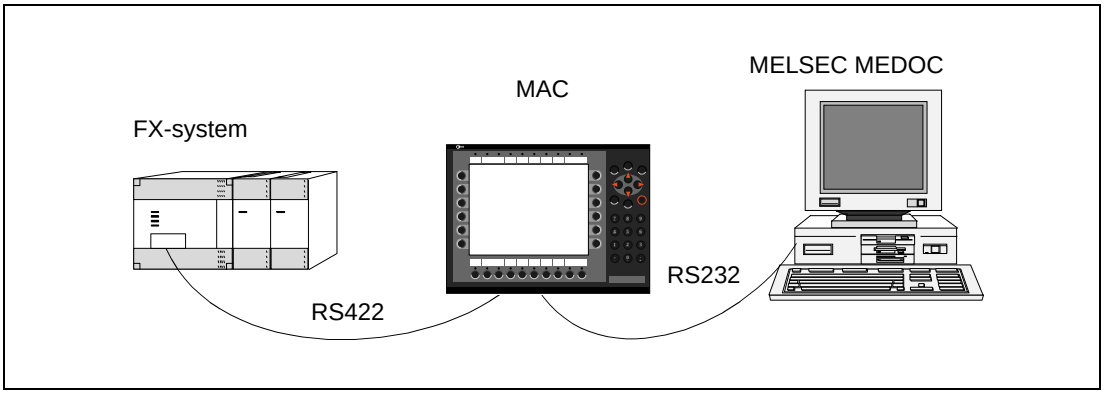


Fig. 15-1: *Terminal with MELSEC MEDOC connected to the programming/printer port*

15.1.2 Settings in the terminal and MELSEC MEDOC

The following settings must be made in the terminal and MELSEC MEDOC for the transparent mode to work.

The terminal

The printer parameters are set in the **Setup** menu under the menu option **Port parameters**.

- The parameters should be 9600 baud, 7 data bits, 1 stop bit and even parity if the terminal is connected to an FX system.
- The parameters should be 19200 baud, 8 data bits, 1 stop bit and no parity if the terminal is connected to one of the A-series communication modules (AJ71).
- The parameters should be 9600 baud, 8 data bits, 1 stop bit and odd parity if the terminal is connected to an A-CPU port.

NOTE Enter the command **BR** on the **command line** under **System signals** in the **Setup** menu.

MELSEC MEDOC

The transfer parameters should only be changed to 19200 baud, 8 data bits, 1 stop bit and no parity when transferring via the A-series communication modules. This is done under **Transfer** in **Setup**. The Interface should also be changed from CPU to AJ71 under Transfer.

The transfer parameters should not be changed in MELSEC MEDOC for other projects. See the Installation manual for cable connections.

The settings also apply for other PC-based monitoring systems such as SCADA.

15.1.3 Connecting two terminals

Two or several terminals can be connected to the same PLC system with transparent mode. The first terminal, terminal 1, is connected to either the CPU port of the PLC system or to the C24 port on a A system via the RS422 port. The other terminal, terminal 2, is connected to the terminal 1 via the RS232 port.

The cables between the units shall be used according to the manual *Installation E900*. Is the distance longer than 15 m a signal amplifier must be used.

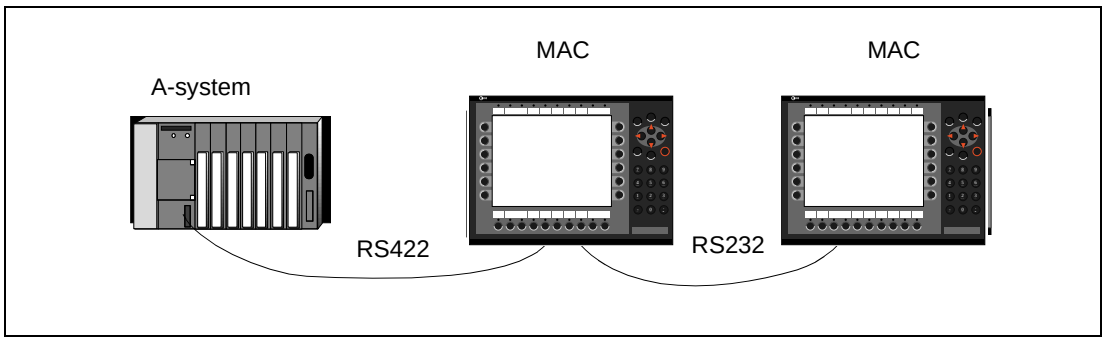


Fig. 15-2: *PLC system with two terminals*

The first terminal is connected in the usual way with the standard cable when connecting two terminals to an FX system or an A-CPU port.

The first terminal is connected according to the instructions in the Installation manual when connecting two terminals to the A-series communication ports.

15.1.4 Settings for two terminals

The following settings must be made in the terminals for the transparent mode to work.

The first terminal

The port parameters are set in the **Setup** menu under the menu option **Port parameters**. The settings under **PLC** shall be the same as when connecting only one terminal to the PLC system. The settings under **Printer/Transparent mode/No protocol** shall be RS232 and 8 data bits and the other parameters can be set freely. In the terminal connected to the PLC system you should define the BR command on the command line (line at the bottom) under **System signals** in the **Setup** menu.

The second terminal

The corresponding settings for **PLC** under **Port parameters**, **Setup** shall be the same as for the first terminal for Printer/Transparent mode/No protocol under Port parameters, Setup. If you use the PLC system FX2N or Q, the command BSD must be entered on the command line under **System signals** in the **Setup** menu.

NOTE

If the two terminals are connected to a Q system, a time-out of 10 seconds must be set in the second terminal. This is made by entering the command T10000 on the command line under System signals in the Setup menu.

15.1.5 Transfer speed

The transfer speed can vary between 600 and 57,600 baud. The remaining parameters should be 8 data bits, 1 stop bit and no parity. The highest transfer speed between the terminals is recommended to obtain the best results. The speed of communication decreases with the number of terminals connected according to the following table.

Number of terminals	Access time to the PLC			
	Terminal 1	Terminal 2	Terminal 3	Terminal 4
1	100%	—	—	—
2	50%	50%	—	—
3	50%	25%	25%	—
4	50%	25%	12,5%	12,5%

Tab. 15-1: Access times when connecting several terminals

15.2 E900 as communication interface (No protocol mode)

The *No protocol mode* is used to connect different PLC systems, connect external units such as bar code readers or weighing scales to the PLC system. The PLC system controls what is to be sent to the communication port. Data coming in to the communication port is written to the PLC register. Communication takes place with the transfer of the data register area according to the control block below.

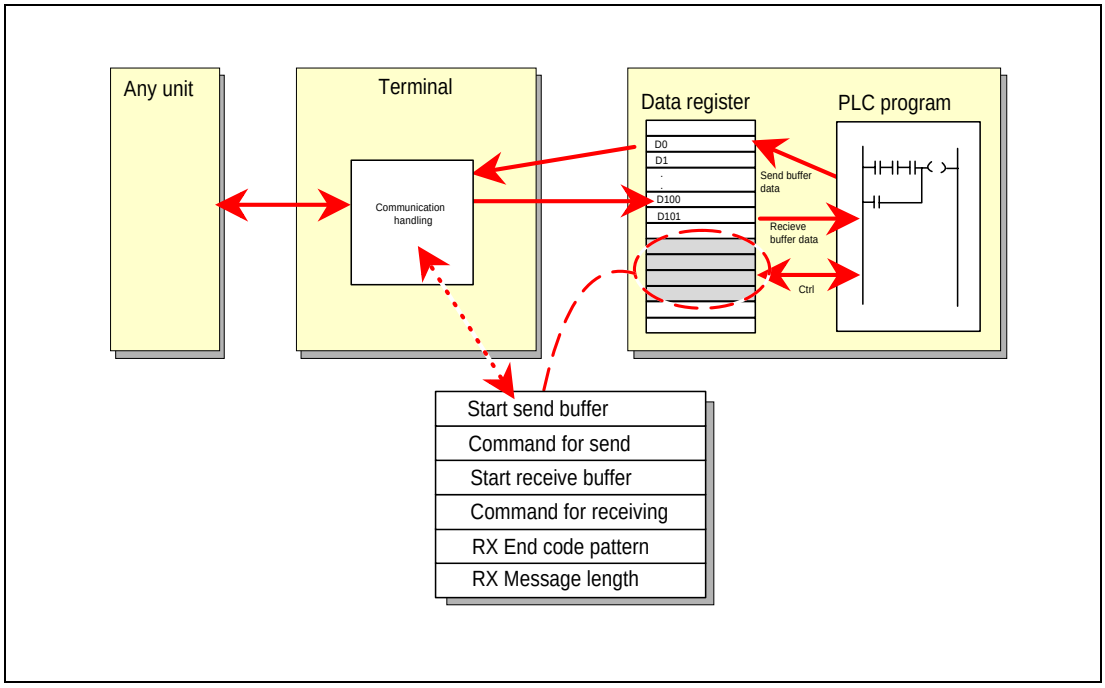


Fig. 15-3: *Connection of different external units*

Register	Description
Dn0	*Start register for the transmission data buffer.* The first register in the buffer area contains the total number of bytes to be transmitted. The following register contains the data to be transmitted. The maximum buffer size is 127 registers = 254 bytes.
Dn1	*Command register for transmission.* Set to 1 by the PLC system when transmission is required. Set to 0 by the terminal when transmission is complete.
Dn2	*Start register for the reception data buffer.* The first register in the buffer area contains the total number of bytes received. The following register contains the data received. The maximum buffer size is 127 registers = 254bytes.
Dn3	• *Command register for reception.* – Set to 0 by the PLC system when it is ready to receive. – Set to 1 by the terminal when the message is available. – Set to -1 (FFFF) on error message (e.g. too short). – Set to 2 by the PLC system when clearing of the port buffer is required. – Set to 3 by the terminal when clearing of the port buffer has been carried out. The port buffer is automatically cleared on start up and when switching between transparent mode and no protocol mode, i.e. the register receives the value 3.
Dn4	*End code* (1 or 2 bytes) on the message received.
Dn5	*Length* of the message received. End code is used if 0.

Tab. 15-2: *Registers*

In the **Setup** menu, **System signals** defines which register will be the first control register in the transfer area. This and the five following registers are used as control registers. The terminal runs in transparent mode if no register is stated.

The PLC system can switch between no protocol and transparent/printer modes in the run-time mode. See the command NP in the system signals section in the chapter *Programming using the PC software*.

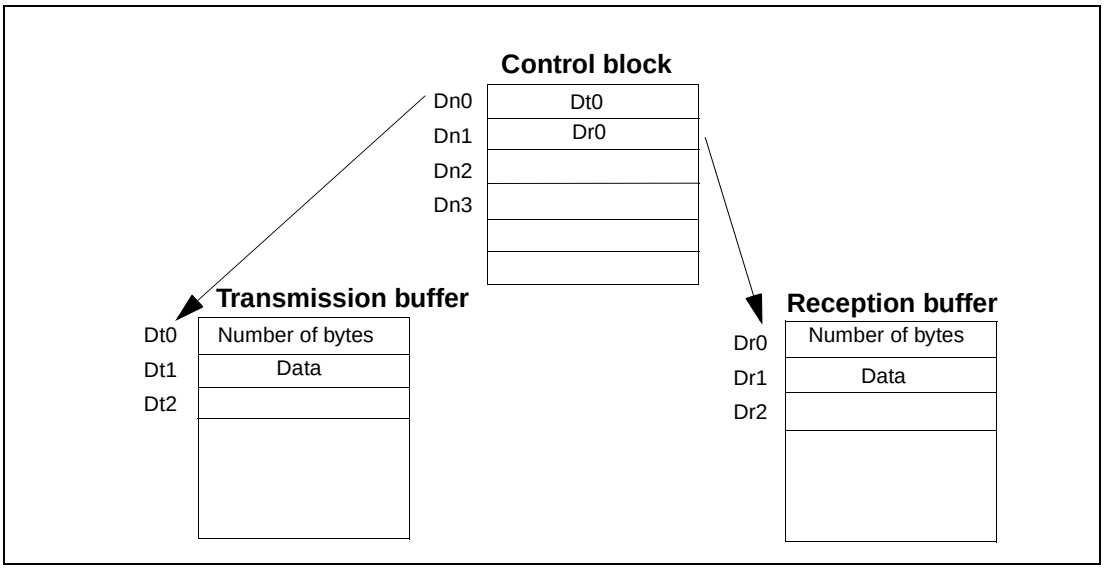


Fig. 15-4: *Control block, transmission- and reception buffer*

15.2.1 Example of using the no protocol mode

The following example describes the use of the no protocol mode for a weighing system. The diagram below shows a three-step block schedule for communication.

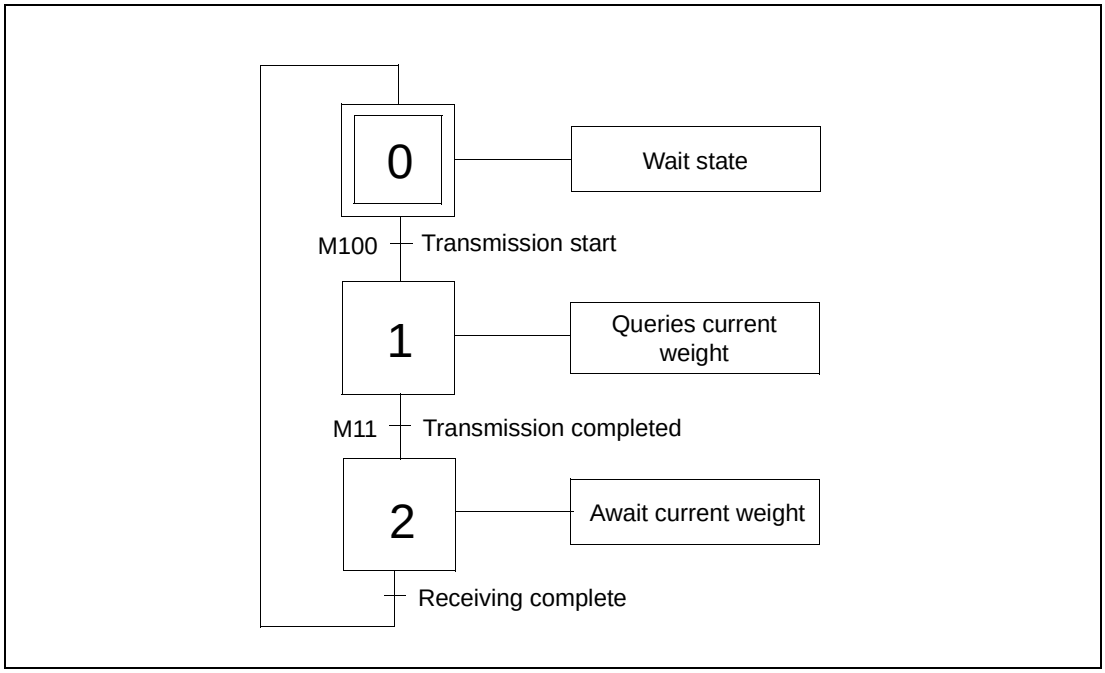


Fig. 15-5: *Three-step block schedule for communication*

The terminal functions as the master towards the weighing system, that is it continually asks the weighing system for the current weight. The diagram below shows the connections between the PLC system, terminal and weighing system.

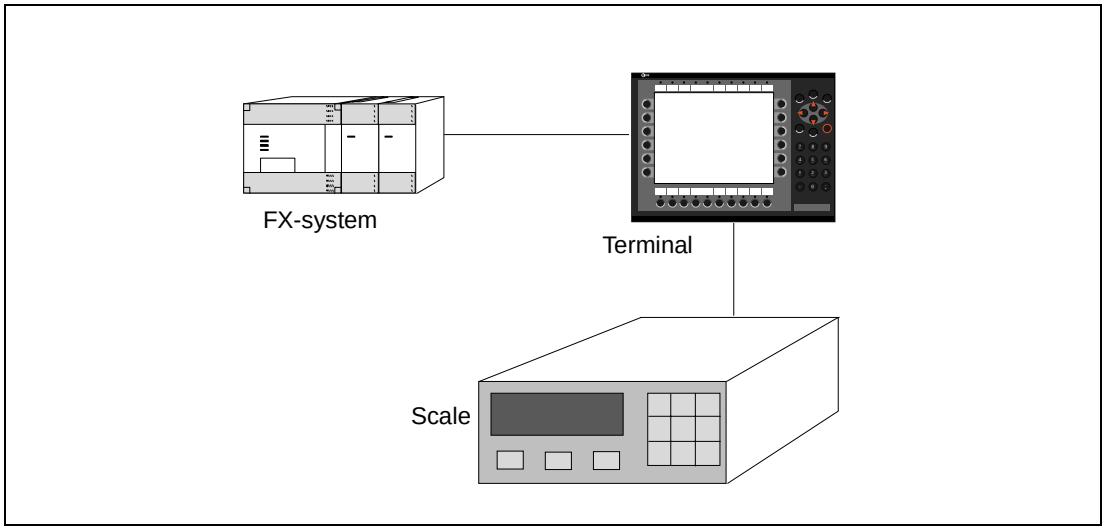


Fig. 15-6: *Connecting the scale and the PLC*

The protocol appears as follows

- Terminal

  | STX | ? | CR | LF |

- Weighing system

  | STX | Weight in kg | CR | LF |

The PLC program appears as follows:

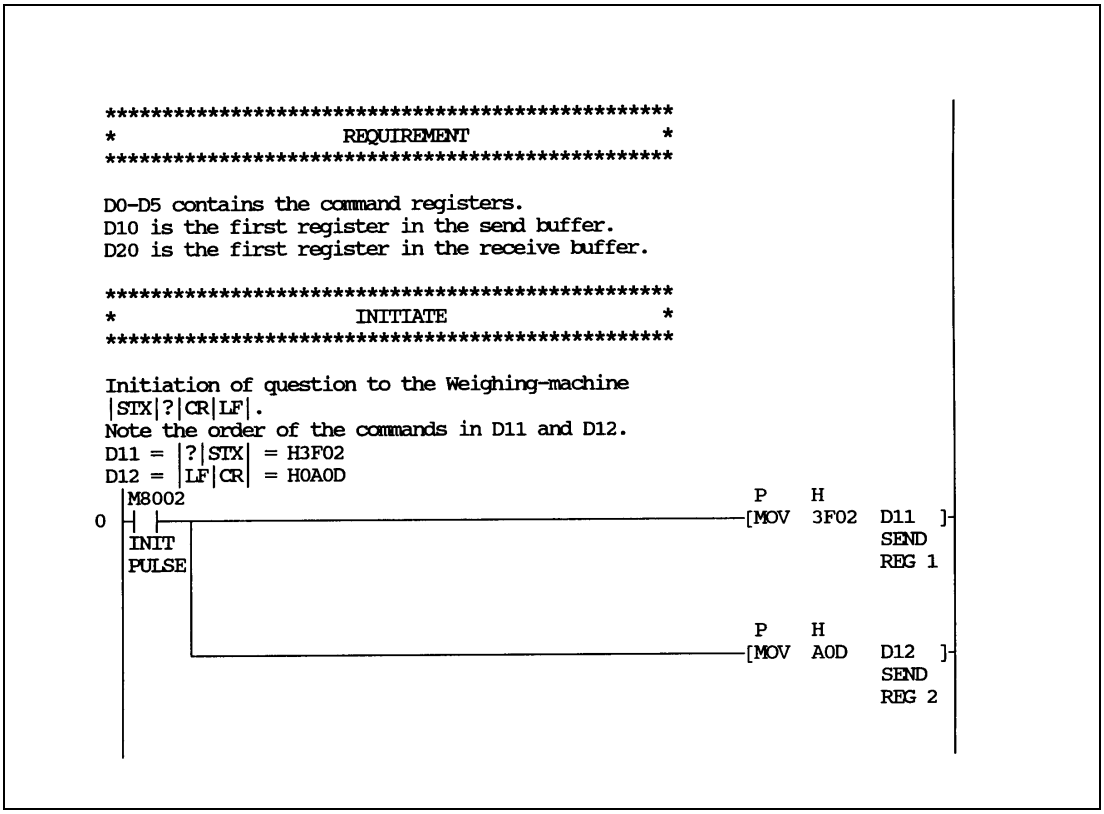


Fig. 15-7: *PLC program (1)*

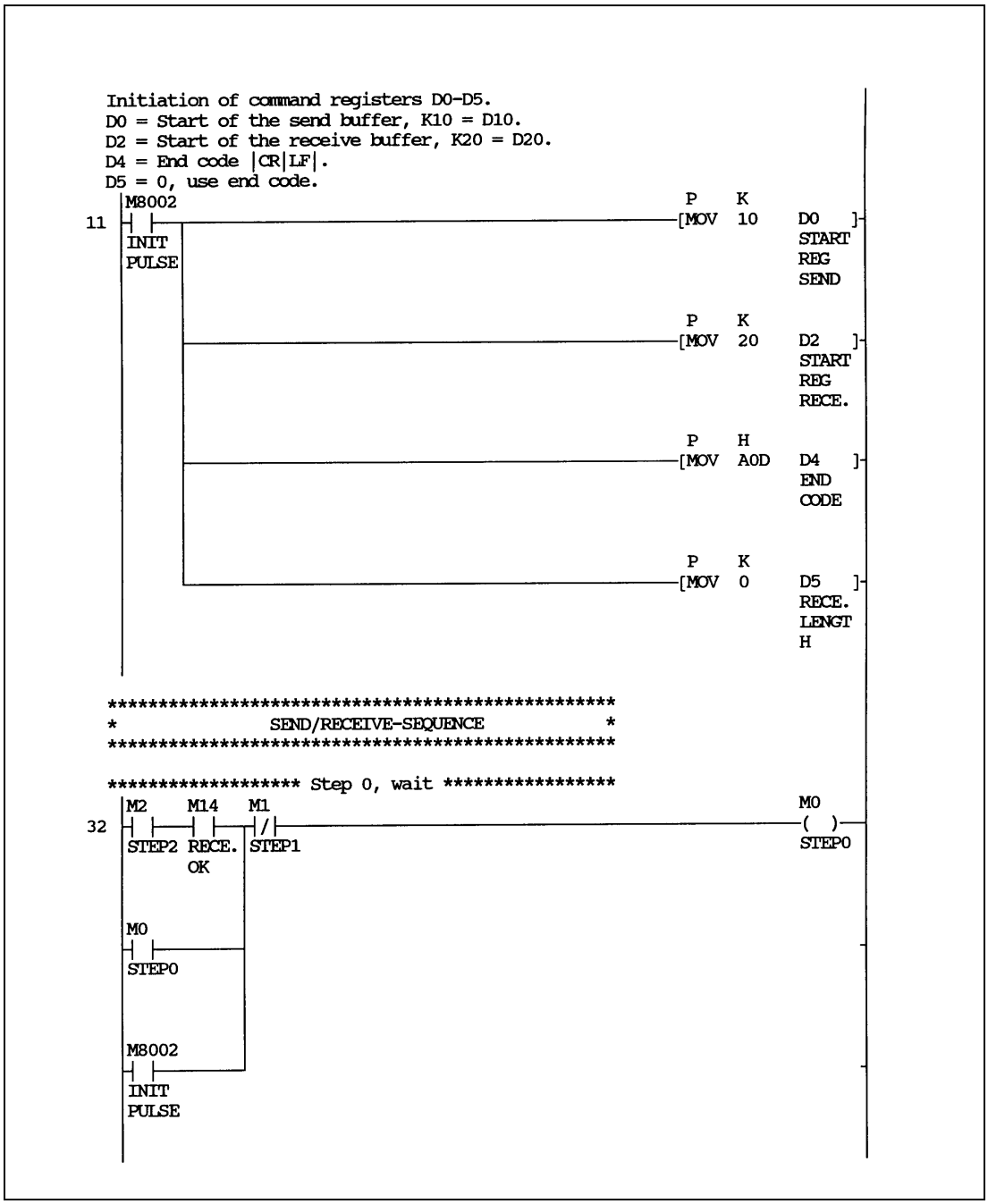


Fig. 15-8: *PLC program (2)*

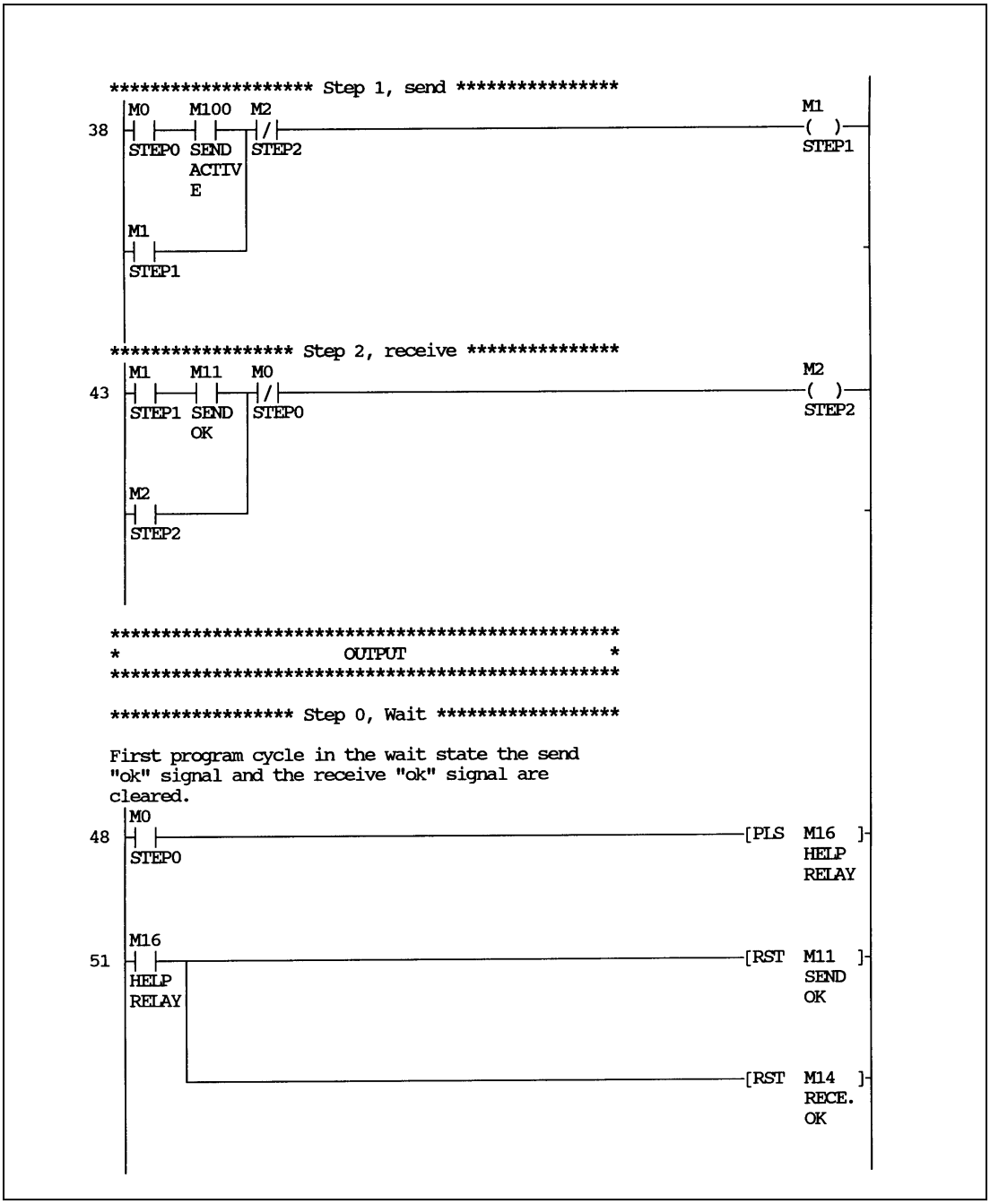


Fig. 15-9: *PLC program (3)*

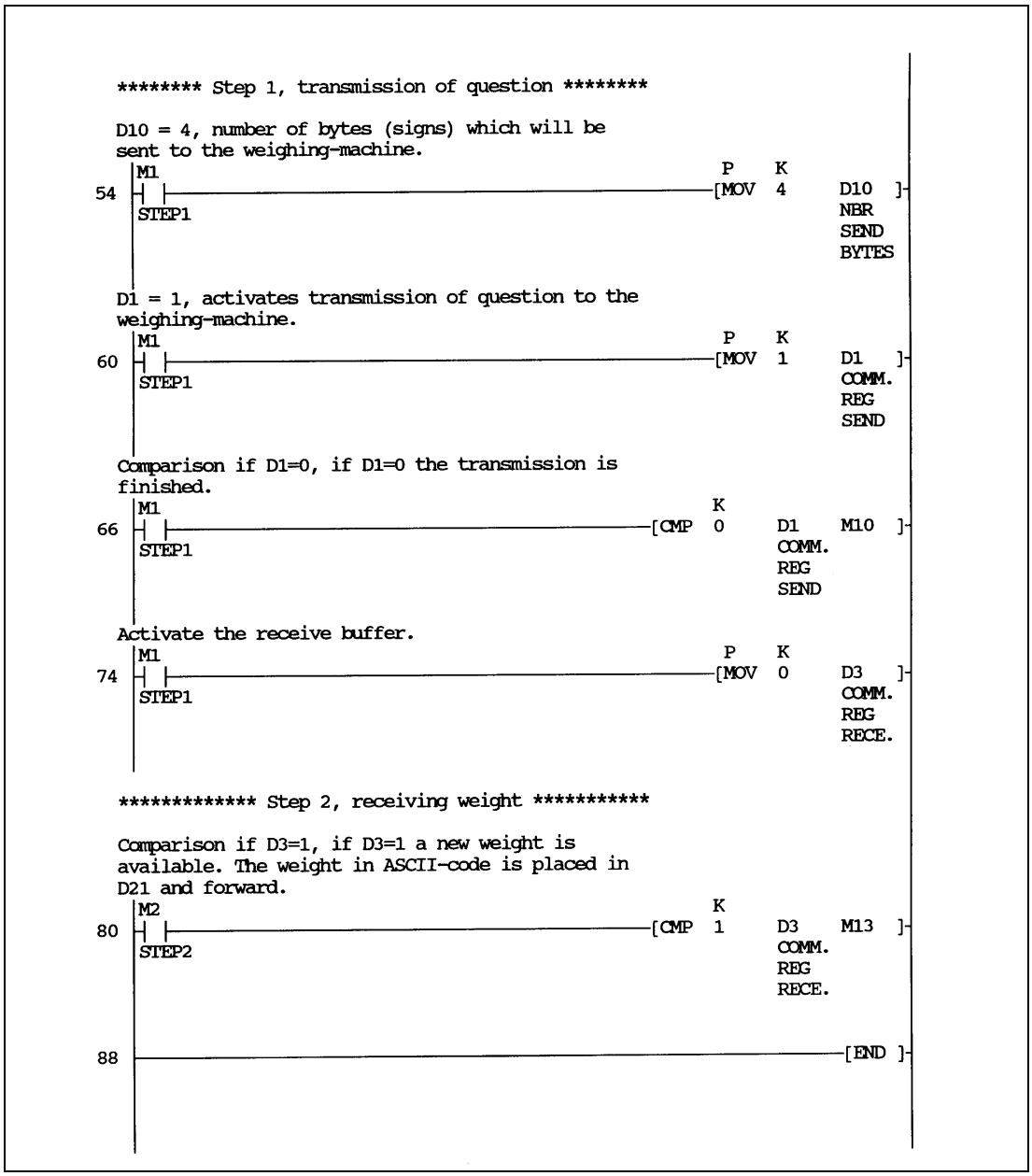


Fig. 15-10: *PLC program (4)*

15.3 Modem connection

A modem can be used for making a connection to a personal computer. The settings for the connection is made in the **Modem Setting** dialog in the **Setup** menu.

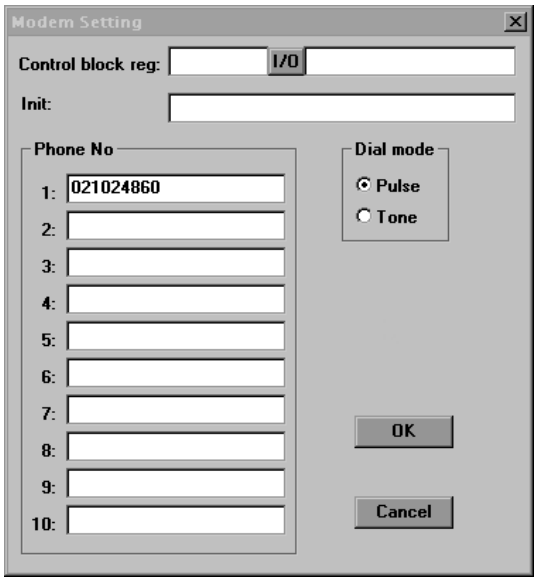


Fig. 15-11: *Modem setting*

Ctrl Block Reg:

The control block occupies two registers. The first register contains the command describing how the PLC system shall call up and establish connection. The second register is used as a status register. This register contains the result of the modem command.

0 Wait for command.
1–10 Call up with the phone number 1–10.
101–110 In the field Phone No you enter Hayes modem commands instead of the phone number. Use the commands 101–110 to enter a init string for each phone number. The init string is sent to the modem.
255 Hang up command.

The status register can contain the following:

- Status codes:

0 Command executed successfully.
1 Dial in progress.
2 Modem established a connection.
3 Modem detected a ring signal.

- Error codes:

101 No connection (when hanging up)
102 Modem detected lost carrier.
103 Unspecified error from modem.
104 Modem gets no dial tone.
105 Busy when dialling.
106 Now answer when dialling.
107 No reply from modem.
255 Unknown error/status.

- Dial mode:
  Select between Pulse and Tone.
- Int String:
  Modem init string
- Phone No. 1–10:
  Complete phone number for call up.

15.4 Expansions card

In the terminal there is a possibility to connect different expansion cards to increase the communication possibilities.

15.4.1 IFC 128E

IFC 128E is an expansion card for connection of external function keys or switches. Up to 128 external function keys/switches can be used. For further information see the manual for the product.

15.4.2 IFC 50E

The expansion card IFC 50E is a communication card for network communication. It is used to connect more than two terminals to one PLC system and maintain good performance. For further information see the manual for the product.

15.4.3 IFC PBDP

With the expansion card IFC PBDP the terminal can be connected to and communicate with a Profibus network. For information about how to setup the card, see the manual for the product. For information about how to configure a Profibus network, see the manual for Profibus network.

15.4.4 IFC MC

With the expansion card IFC MC you can connect a PCMCIA card to the terminal. You can select to use the memory card to expand the project memory, use it as backup or for moving files between terminals. For further information we refer to the manual for IFC MC.

15.4.5 IFC PI

With the expansion card IFC PI you can connect a parallel printer with HP-PCL5 language to the MAC E900. For further information see the manual of the product.

15.4.6 IFC GA

With the expansion card IFC GA you can connect the MAC E900 via the A76T-BUS module to communicate with the MELSEC A-BUS. For further information see the manual of the product and the manual of the A-series.

16 Project transfer

The project must be transferred from the PC to the terminal, in order to be used in the terminal.

Connect the PC with MAC Programmer+ to the terminal via the cable MAC-PC-CAB-R2 or MAC-PROG/9-CAB.

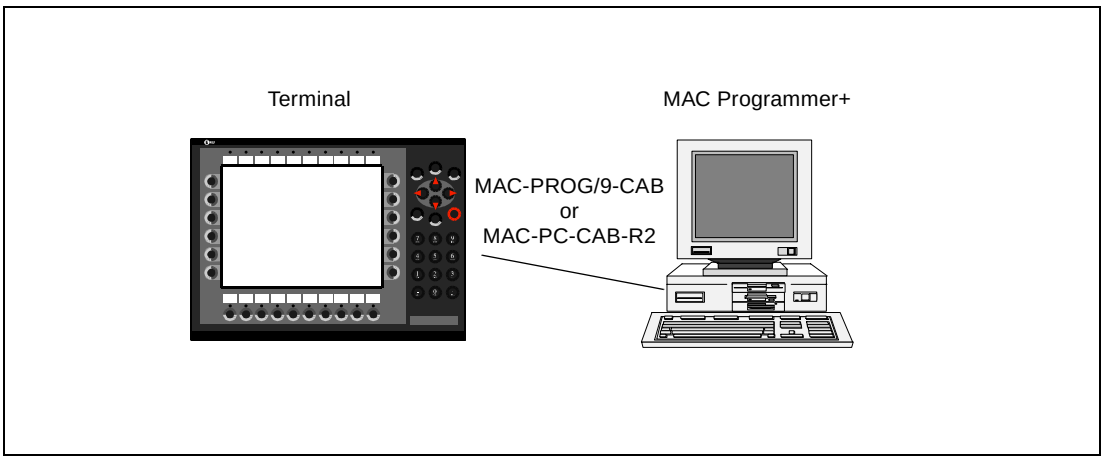


Fig. 16-1: *Connection between PC and terminal*

16.1 Terminal settings

The transfer parameters are set in the terminal in configuration mode under **Setup, Port Parameters, MP.**

NOTE The communication settings must be the same in the PC software as in the terminal.

16.2 Transfer settings

The transfer is controlled from the PC software. In the menu Transfer you can select what is to be transferred. In the PC software you make the settings in the Transfer settings menu.

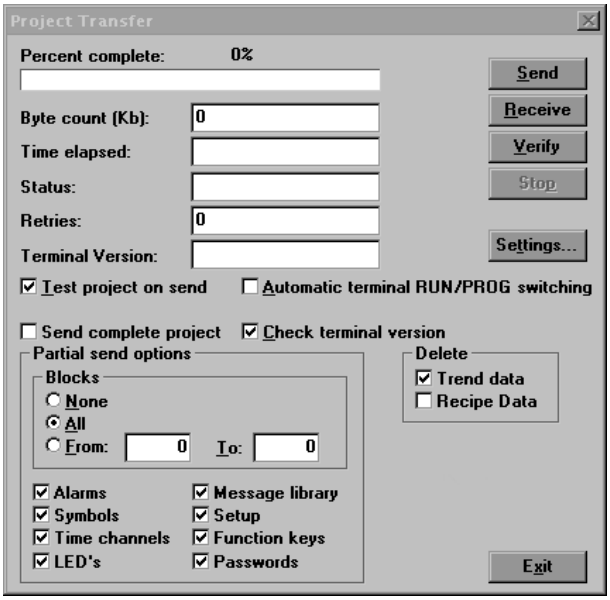


Fig. 16-2: *Dialog box project transfer*

Parameter	Description
Percent complete	Shows the status of the transfer during tranfer.
Byte count	Shows the number of bytes transferred during transfer.
Time elapsed	Shows elapsed time since one of the function´s send, receive or verify was activated.
Status	Shows which part of the project that is transferred. For example Setup, block, alarm groups, symbols and function keys.
Retries	When the transfer fails, the PC software makes a number of retries before the transfer is shut down.
Terminal version	When contact is enabled with the terminal, current terminal type and program version are displayed here.
Test project on send	Tests the project during transfer.
Automatic terminal RUN/ PROG switching	The terminal is automatically put in transfer mode and will return to the previous mode after transfer.
Send complete project	Select if you want to transfer the whole project.
Check terminal version	Checks the terminal version and gives a warning if the project and the terminal don´t match.
Partial send options	None: No blocks are sent to the terminal. All: All blocks are sent to the terminal. From To: State an interval of blocks to be sent to the terminal. Alarms: Select if alarms shall be sent to the terminal. Symbols: Select if symbols shall be sent to the terminal. Time channels: Select if time channels shall be sent to the terminal. LED´s: Select if LED´s shall be sent to the terminal. Message library: Select if message library shall be sent to the terminal. Setup: Select if setup shall be sent to the terminal. Function keys: Select if function keys shall be sent to the terminal. Passwords: Select if passwords shall be sent to the terminal.
Delete	Trend data: Delete all stored trend data in the terminal. Recipe data: Delete all stored recipe data in tht terminal.
Send	Transfer the project to the terminal according to current settings.

Tab. 16-1: *Parameter settings for project transfer*

Parameter	Description
Receive	The PC software loads the project in the terminal. Current project in the PC software will be overwritten.
Verify	Compares the project in the PC software with the project in the terminal.
Stop	Interrupt the transfer.
Settings	Transfer parameter settings. Must be the same as in the terminal. See the following table.

Tab. 16-1: *Parameter settings for project transfer*

The Setting menu

Parameter	Description
Baudrate	Baudrate (bits/second).
Timeout	Number of milliseconds between retries.
Retries	Number of retries before the transfer is shut down.
Parity	Select type of parity control.
Databits	Select number of data bits. Must be 8 bits.
Stopbits	Select number of stop bits.
Speed set manually	Used when communication via a modem. You must set the baudrate in the terminal and PC software manually and they must be the same.

Tab. 16-2: *Setting menu*

NOTE

Communication can fail if other programs are running under Windows at the same time as the transfer takes place. Failure is avoided by shutting down the other programs.
During block transfer, links to relevant symbols are included. If the symbols are not transferred, problems may occur.

16.2.1 Transfer via modem

NOTE

If **Automatic terminal RUN/PROG switching** is used the baudrate of the PC must be set to 2400 baud.

If another baudrate than 2400 baud is used the parameter **Speed set manually** under **Comm Settings** in the **Transfer** menu must be changed and the corresponding baudrate must be set for the terminal.

The following modem commands are recommended.

- AT &D0 &K3 &W

17 Passwords

This section describes how to use security levels and passwords in the terminal. This function makes it possible to create a security system for the terminal. The operators can be given different accessibility to the system.

Blocks, function keys and maneuverable objects can be allocated a security level. Each security level is given a password. The user must log in with a password for the security level concerned or a higher one to gain access to the various levels. It is not essential to use this function.

17.1 Defining security levels

Blocks, function keys and maneuverable objects can, when they are defined, be allocated a security level. A security level (0 to 8) is given in the dialog box for the object. Everyone will have access if the 0 security level is used.

17.2 Defining passwords

The password is defined for security levels 1 to 8 under **Passwords** in the **View** menu.

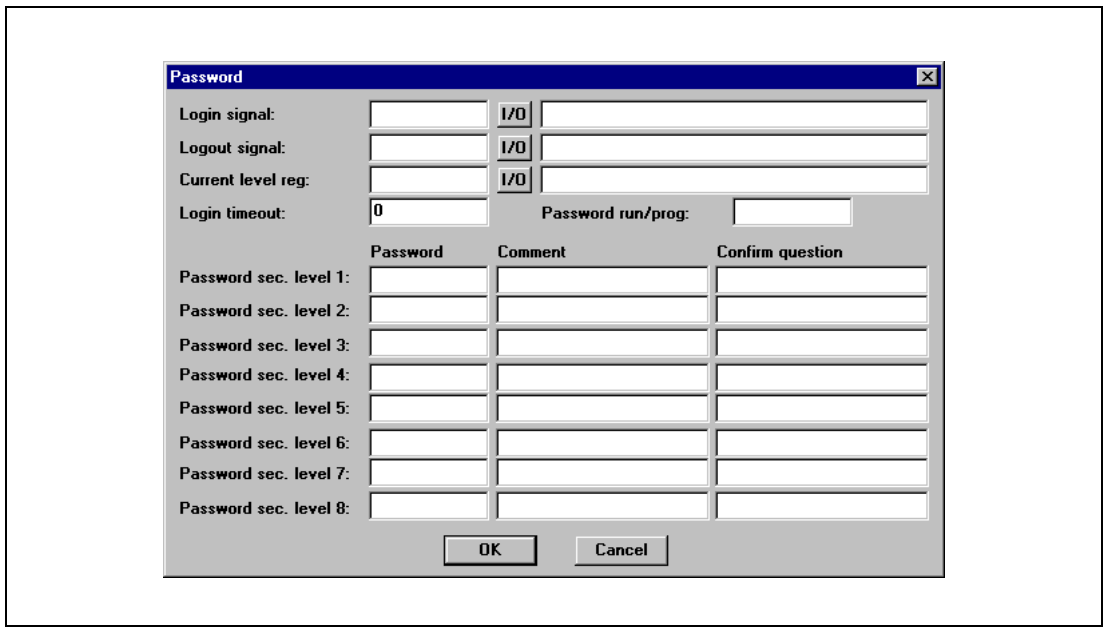


Fig. 17-1: *Dialog box password*

Parameter	Description
Login signal	State here the digital signal which when set generates the entry field for logging in. The entry field for logging in can also be linked to a function key, see the *Function keys* section.
Logout signal	State here a signal which when set logs out the current user. This function can also be linked to a function key, see the *Function keys* section.
Current level reg	State a register where the terminal put the current password level (0-8).
Login timeout	State here the time in minutes during which the terminal can remain unused before the user is automatically logged out. If 0 is used no logging out takes place.
Password RUN→ ROG	Here you can state a password which must be entered when switching from RUN to PROG manually. The function is not used when switching from PROG to RUN or when Auto RUN/PROG switching from the PC software is used.
Password sec level 1-8	Enter the password and any comments for security levels 1-8.
Confirm question	Enter a question that should be confirmed by the operator before maneuvring is enabled.

Tab. 17-1: *Parameter settings for passwords*

17.3 Logging in

Logging in is either controlled from a function key or through a digital signal from the PLC system. The entry field for logging in is shown if the function key linked to the function LOGIN is pressed or the digital signal is activated. State the password here. The password is linked to the security level, see above.

17.4 Password for transfer

On the command line under System signals in the Setup menu you can state the command PDxxxxxxxx. The operator then have to enter a password to have access to the functions in the Transfer menu.

17.5 Password for all terminal functions

On the command line under System signals in the Setup menu you can state the command PSxxxxxxxx. The operator then have to enter a password to have access to the functions in the terminal.

17.6 Altering passwords in run-time mode

The password can be changed in run-time mode with the LOGSET function, see the *Run-time Mode* section. The LOGSET function must be linked to a function key, see section 18.9, *Function keys*.

18 Run-time mode

The function of the various objects in run-time mode is described in this section. These do not respond to any menu option in run-time mode. Block 0 is automatically shown on the display when switching to run-time mode.

In E900 the built-in key board is used to select and change values in run-time mode.

An error message is shown on the display if a communication fault should occur between the terminal and the PLC system. The terminal starts automatically when communication restarts.

18.1 Selecting maneuverable objects

The arrow keys are used to jump between maneuverable objects. The choice of object is according to the following principle: The cursor's position is assumed to be the center of a cross. Press the right arrow and the first object found in area A according to the diagram below is selected. Should the system not find an object in the narrow field on the right, it searches in area a. The down arrow searches in areas B and b, the left arrow searches in areas C and c and the up arrow searches in areas D and d.

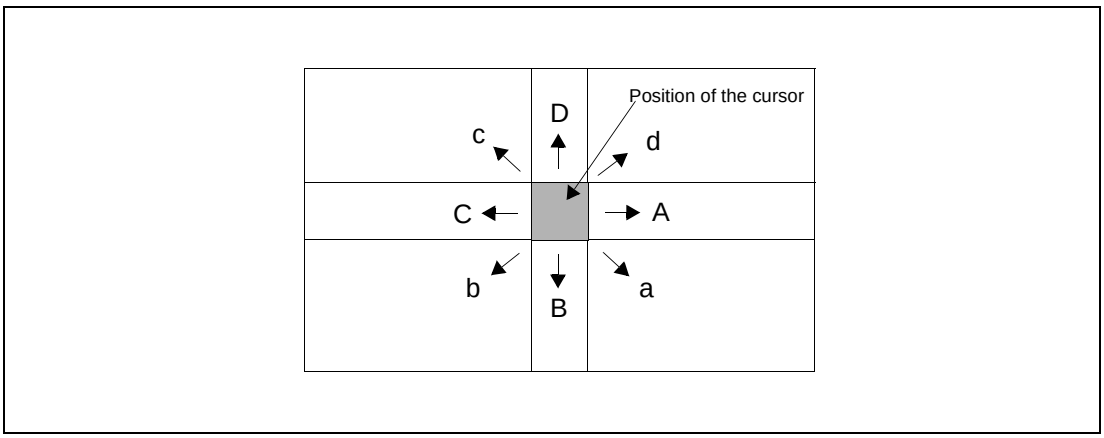


Fig. 18-1: *Choice of objects*

18.2 Maneuvering Graphicblocks

Move between the maneuverable objects with the arrow keys. Objects selected are highlighted with a blinking frame.

18.2.1 Digital objects

Digital objects, text, symbol and filled objects, switch status when [ENTER] is pressed. If the INC and DEC functions have been linked to function keys, the signal linked to the object with these keys is set and reset respectively.

18.2.2 Analog objects

ASCII objects

Mark the object and enter the required text. Finish writing with [ENTER].

Multichoice objects

Place the cursor on the object and press [ENTER]. An option list is now shown with all the maneuverable states. Move stepwise to the required state and press [ENTER].

Numerical objects

To maneuver numerical objects write in the value and press [ENTER].

Numerical table objects

When the table object is marked press [ENTER] to select the first item in the table. Use the arrow keys to move the cursor between the items in the table. Change the value in the marked item and press [ENTER].

Potentiometer objects

The object is maneuvered with the arrow keys by placing the cursor on the object and pressing [ENTER]. The value can now be increased/decreased with the number that corresponds to the setting ”Scale Ticks” in the object. End maneuvring with [ENTER].

The object can also be maneuvered with the INC and DEC functions on the condition that they are linked to the function keys, see section 18.9, *Function keys*.

Bar object

The min/max indicators for bar objects can be set at the current value by pressing [ENTER].

The max/min indicators are reset when you touch the bar.

Trend objects

Trend curves can show historic data in the run-time mode. Mark the required trend curve and press [ENTER]. A dialog square is now shown and the date for the data to be presented can be chosen. ”History” is shown at the bottom of the square. Press [ENTER] again to return to real-time display. Trend data is stored in the files whose names are stated when the trend object is defined.

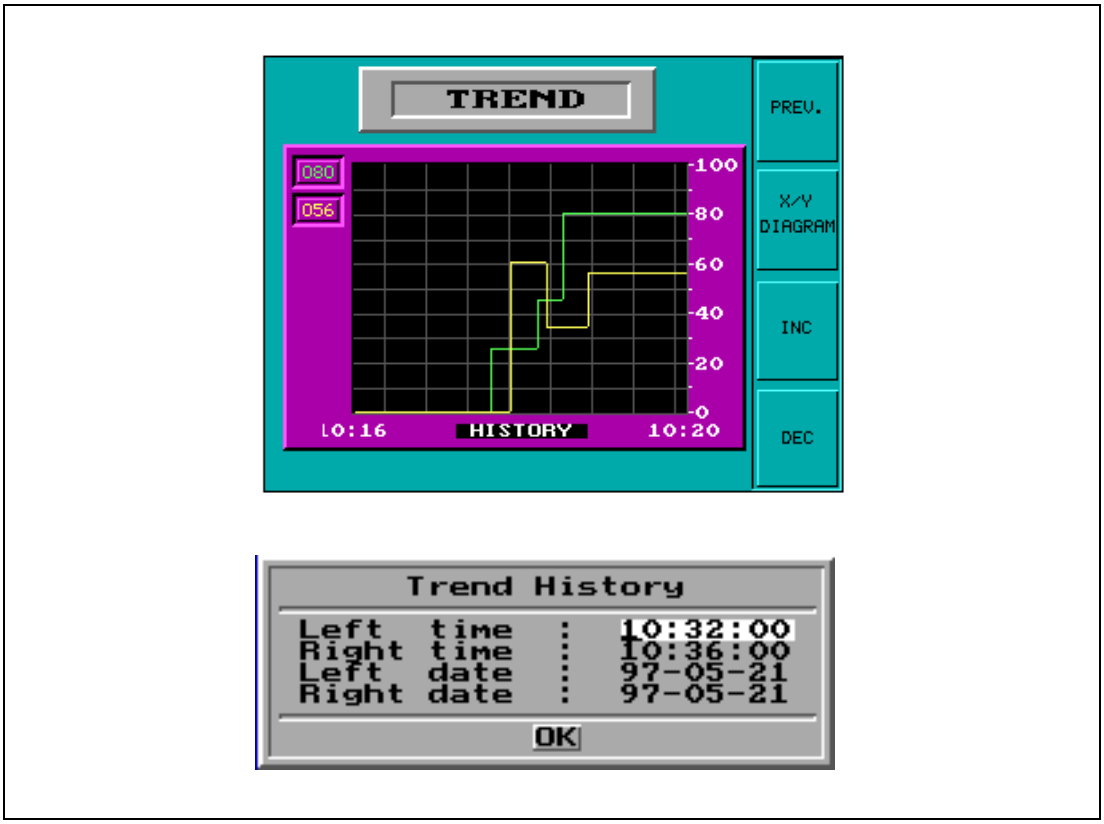


Fig. 18-2: *Trend object*

Diagram object

The curves in the diagram can be edited in run-time mode. Select the object you want to maneuver and press [ENTER]. Select curve and press [ENTER]. Now a flashing cursor is shown on the selected curve. Select coordinates on the curve with the arrow keys left and right. Press [ENTER] at a point and the cursor stop flashing. It is now possible to change the coordinates for the curve with the arrow keys. Press [ENTER] to save the changes and [PREV] to return without saving. Leave the editing mode by pressing the [PREV] key.

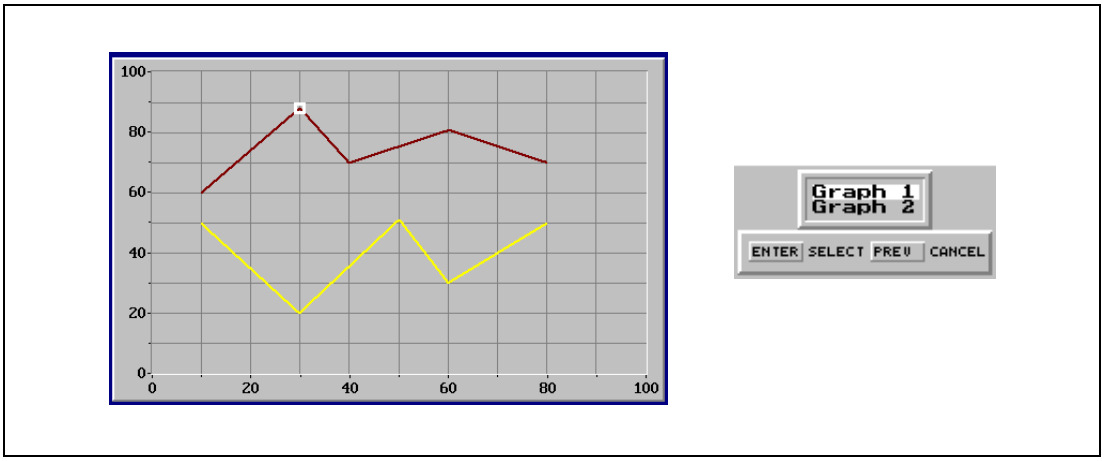


Fig. 18-3: *Diagram object*

18.2.3 Other objects

Digital clock

The digital clock can be altered by marking the object and giving the required time. Finish with [ENTER].

NOTE

If you use the clock in the PLC system and you want to change it, this must be done in run-time mode.

Jump object

Mark the object and press [ENTER].

18.3 Maneuvering Textblocks

Textblocks consist of text lines with static and dynamic objects. The dynamic objects show the current status for those signals to which the object is linked. Certain dynamic objects are maneuverable which means that you can change their status and values in run-time mode.

To alter a maneuverable object you use the arrow keys to move the cursor to the right position so that the object is marked.

Form feed is automatic after 60 lines when printing out a Textblock if nothing else is stated during configuration under printer setting. The printer must be IBM compatible and handshaking takes place through XON/XOFF or CTS/RTS protocol which is selected during configuration. It is possible to print out text block on a laser writer, e.g. HP PLC5.

18.3.1 Digital objects

Digital objects are maneuvered by marking the required object. Then press [ENTER] to alter the status of the object.

18.3.2 Analog objects

Analog and Date/Time

The objects are maneuvered by placing the cursor on the right object. After this the new value is written in. Finish with [ENTER].

Before [ENTER] is pressed, new value entry can be aborted by leaving the field with [↑] or [↓]. The old value will then be retained.

Text objects

Maneuver a text object by moving to the object and pressing [ENTER]. Then an entry field is shown. The entry field is either on the first or last line depending where the object is on the display. Scrolling of the entry field takes place if the text is longer than the width of the display. Press [ENTER] when writing is complete.

Multichoice objects

Maneuver a multichoice object by moving to the object with the arrow keys and press [ENTER]. An option list is now shown with all maneuverable states on the display. Move to the required state, press [ENTER] and the three digital signals linked to this state are forcibly set.

Jump object

Move to the object and press [ENTER].

18.4 Alarms

Alarms are shown in the textblock with the text ■ALARM■ in the upper right-hand corner of the display (you cannot select this during configuration). A red bell blinks in the upper right-hand corner of the display for alarms in Graphic blocks.

The alarm is presented in an alarm list with the predefined alarm texts. The alarm list contains the latest alarms which are listed according to alarm group according to definition and with the latest alarm at the top of the list. The alarm list's size in number of kbytes is selected during configuration. The alarm list is shown on jumping to the alarm block. The status for the alarm is shown for each alarm as well as the time when the alarm was activated, deactivated or acknowledged depending on the selected display mode.

Alarms can have the following status.

Symbol	Status
*	Active, Not acknowledged
$	Not active, Not acknowledged
-	Active, Acknowledged
<blank>	Not active, Acknowledged

Tab. 18-1: *Alarm status*

Alarm can be shown in the following format.

Display format	Description
S	Time when the alarm was activated
E	Time when the alarm was deactivated
A	Time when the alarm was ackknowledged

Tab. 18-2: *Alarm format*

To get to the alarm block you can either press [LIST] or define a jump to system block 990 in a Textblock or allow the PLC system to produce a list via the Display signal for block 990.

To acknowledge an alarm place the cursor on the line where the alarm is and press [ACK]. Alarms can be printed out directly as they occur or change status if a printer is connected. This should be stated when the alarm is defined. The alarm is printed out with date, time, status and the alarm text.

Return to the previous block by pressing [PREV] on the terminal.

The current content of the alarm list can be printed out by giving the Printer signal for block 990.

18.4.1 Block linked to alarms

Text or graphic blocks can be linked to alarms. This means that when the operator press the INFO key at an alarm in the alarm list a block linked to the alarm is shown. This block can contain information about the alarm and reccomended actions. It is only possible to press the INFO key if current alarm is linked to a block.

18.5 Graphic alarm page in run-time mode

The page is graphic and can be modified by the user. The function keys can be linked to the functions in order to enlarge or reduce the text on the alarm page as well as the function for flipping through the pages. Furthermore there is the possibility to select the showing of date and time.

Alarms can be sorted into groups and you can select which group is to be shown. Status is shown with different colors, the colors defined when setting the alarm group. In E900 you select an alarm group with the arrow keys left and right.

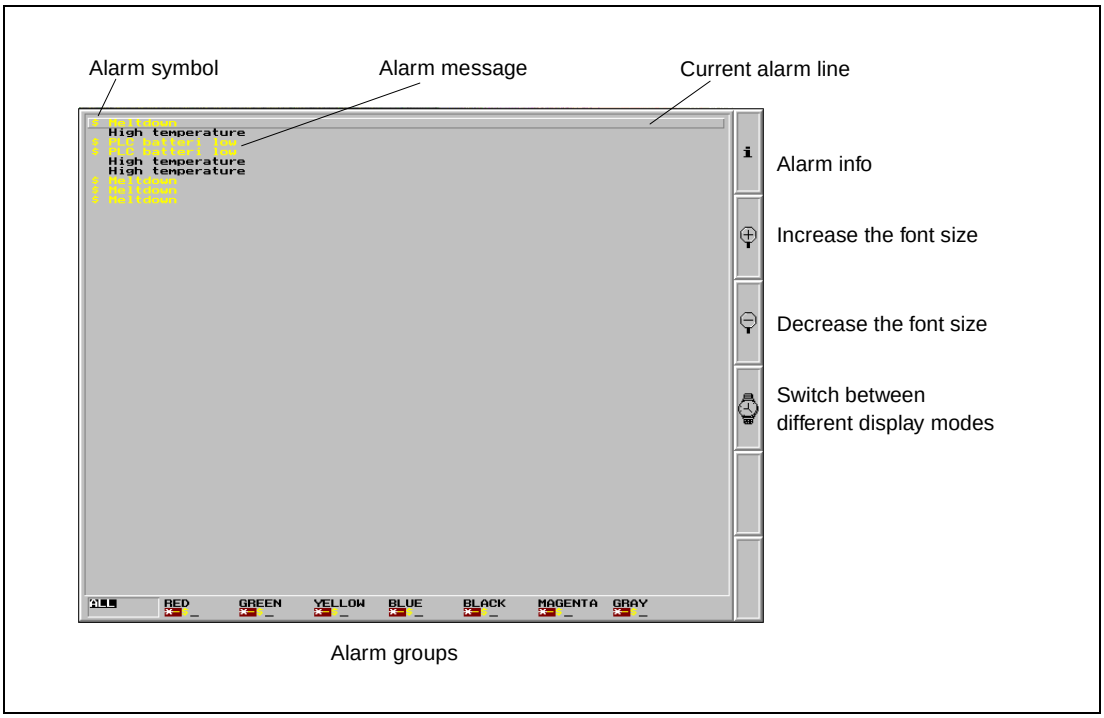


Fig. 18-4: *Graphic alarm page*

18.6 System monitor

The system monitor is a block where the operator can view or change the value for PLC signals in run-time. The value is presented as a decimal, hexadecimal or ASCII value.

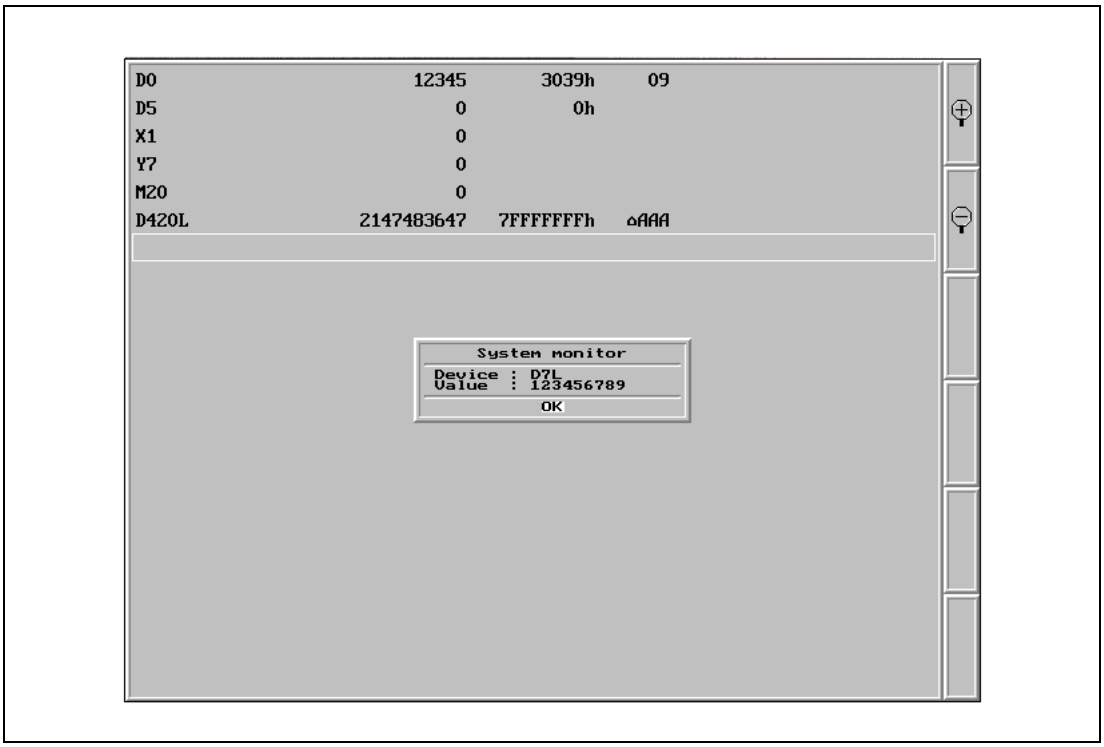


Fig. 18-5: *System monitor page*

18.7 Time channels

The page with the time channels is shown when system block 991 is activated either through a jump object or a digital signal linked to the block. The time channel's values can be read and/or changed. The option Time channels in the Online settings menu must be set to YES in order to alter the time channel's value in run-time mode.

To read or change values for a time channel you move the cursor to required line and press [ENTER]. Press [DONE] to exit from the time channel's definition. Exit from the time channel menu with [PREV]. The block from which the time channel block was activated is then shown.

18.8 Passwords

Passwords can be altered in run-time mode with the LOGSET function. When the function key linked to LOGSET is pressed a dialog is shown where the passwords for each security level can be altered.

NOTE It is not possible to state the security level for block 0.

18.9 Function keys

The digital signal linked to a function key is activated according to the function linked to the key when it was defined.

18.10 LEDs

The LEDs light according to the value of the analog register which is linked to the LED.

19 Using the keyboard

This section describes how the terminal is constructed as well as how the built-in keyboard is used.

19.1 Modes

The terminal has three modes:

- programming
- run-time
- configuration

Switch between programming, PROG, and run-time, RUN, by simultaneously pressing the [←] and [MAIN] keys. The possibility to switch between PROG and RUN from the terminal can be controlled via a password defined under Password settings. For more information see chapter 4, *Programming using the PC software*.

The setup mode is reached by selecting PROG and pressing any key when the start-up screen is shown.

The basic settings for such things as choice of PLC system, menu language, character set and date and time format are made in the configuration mode.

The programming mode is the mode where the application is constructed. It is here you create text- and graphic blocks, define the time channels, alarms, recipes and function keys. We recommend that the terminal is programmed using the PC software.

NOTE Block 0 must be defined in the project in order to be able to run the application.

The application is monitored and run in the run-time mode. You can see how the various objects change and also maneuver certain objects. This assumes that the terminal is connected to a PLC system.

19.2 Keyboard

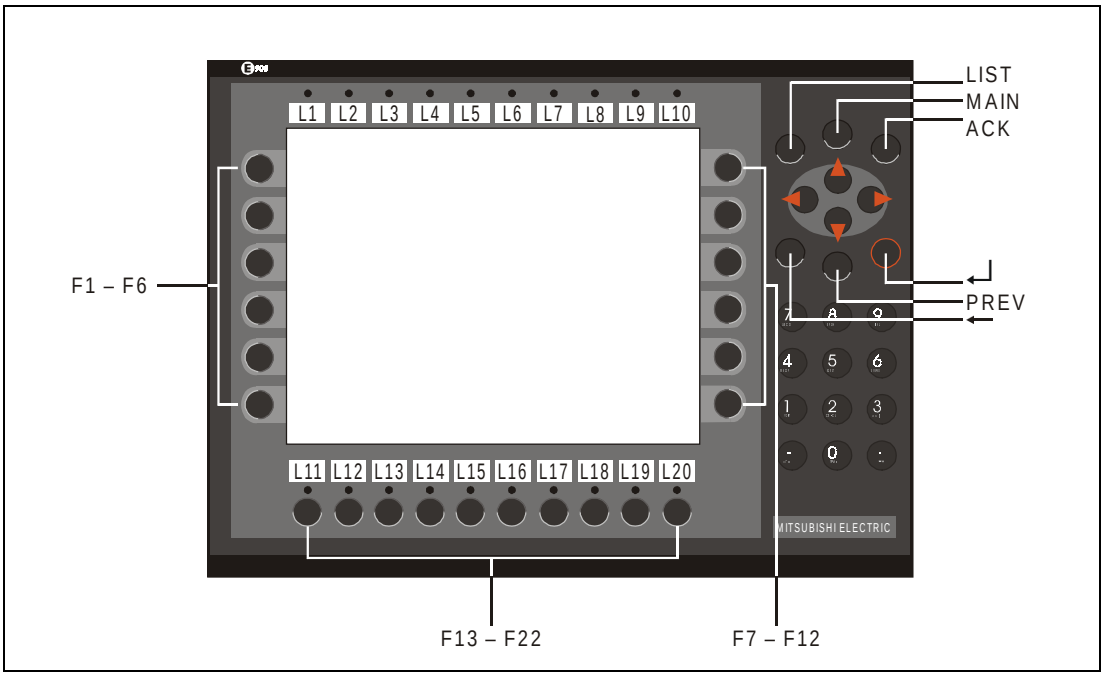


Fig. 19-1: *Function keys and key bowls*

Numerical keys

The numerical keys have several functions, A to Z as well as special characters which are reached by pressing 1 to 5 times on a key. The time interval between the pressings can be set. If the key is not pressed within the time interval the cursor is moved to the next prosition. The keys C1 to C4 are used for national characters.

Arrow keys

The arrow keys are used to move the cursor within a menu or dialog.

Built-in function keys

The [↵] key is used to confirm settings made and go on to the next line/level.

The [PREV] key is used to return to the previous block.

The [LIST] key is used to bring up the alarm list.

The [ACK] key is used to acknowledge an alarm in the alarm list.

The [MAIN] key is used to jump to block 0 in Run-time mode.

The [←] key is used to delete the character to the left of the cursor.

19.2.1 Key sequences

There are three key sequences in the terminal with the following functions.

Key sequence	Function
[←] [MAIN]	Switch between PROG and RUN.
[←] [F1]	Hold this combination pressed during start up to activate the mode for downloading of system programs, see the *Appendix* section.
[←] [PREV]	Calls up the information page, see the *Information page* section.
◄ ►	Hold this combination pressed during start up to activate the self test function.

Tab. 19-1: *Key sequences*

19.3 Help line

There is a help line in the terminal which contains functions linked to the function keys located directly under the line. The functions of the keys changes depending on which level is active. The help lines are only shown in the programming mode.

Function	Description
NEXT	Used to go to the next level. Settings made are saved.
LEAVE	Return to the previous level.
TOGGLE	Switch alternately.
LIST	Show option list.
INSERT	Insert
DELETE	Delete
HELP OFF	Switch between showing and removing help line.
SELECT	Mark an object.
SNAP OFF/ON	SNAP ON means moving and size change in 8-pixel steps. OFF = 1-pixel steps.
NEWOBJ	Create a new object.
PREVOBJ	Show previous object.
NEXTOBJ	Show the next object.
EDITOBJ	Edit an object.
DELOBJ	Delete an object.
MOVE	Move an object.
SIZE	Change the size of an object.
ENLARGE	Increase the character size in an object.
REDUCE	Decrease the character size in an object.

Tab. 19-2: *Help line*

19.4 Menu system

The terminal is based on a menu system which makes it easier for the user to write in to and monitor his system. The following general rule applies when selecting from menus:

Mark an object by moving the cursor to the object with the arrow keys and press [Next] in the help line. You can scroll the text with the arrow keys if the Text-block holds more information than can be shown on the display. Press [LEAVE] in the help line to exit from a menu.

19.5 Editing graphic objects

The arrow keys are used when editing a graphic object to change the size of or move the object in 8-pixel steps. Changing/moving pixelwise is done by pressing [SNAP OFF] in the help line.

To move stepwise through all the objects in the editing mode press the [SELECT] key followed by [NEXTOBJ] in the help line. The object marked is inverted and the object type is shown in a square.

When an object is marked press [EDITOBJ] to edit the object's parameters, [DELOBJ] to erase the object and [LEAVE] to return. Point to an object and press [MOVE] or [SIZE] and make the changes with the arrow keys to move or change the size of the object.

Set the real time clock

The real time clock in the terminal is set under **Setup**, **Date/Time**.

Select the alternative Set Terminal clock. The date and time is now displayed. Press [SET] to change the settings. Enter required date and time. In the editing mode you move the cursor with the arrow keys. Press [NEXT] to return or interrupt the editing before [ENTER] is pressed.

20 Connection to an Allen Bradley PLC system

This chapter describes how to connect an Allen Bradley PLC system to the terminals in the E series and how they communicate. Addressing of an item in the Allen Bradley PLC system is done in the normal Allen Bradley way. It is not possible to maneuver outputs in some PLC systems because they are write protected. For information about Allen Bradley systems we refer to the documentation for the current system.

The terminal supports the following PLC systems

- PLC5 supporting the DF1 commands Protected read/write with 3 address fields.
- SLC 500 series
- MicroLogix 1000 series

20.1 Settings in the terminal

For communication with Allen Bradley PLC systems the following settings must be made in the terminal.

Project settings

When the project is defined you select which PLC system the terminal should be connected to. Select Allen Bradley DF1 in the **project settings** if you will use an Allen Bradley PLC system.

Port parameters

Select **Port parameters** in the **Setup** menu.

Parameter	Settings
Port	RS422/RS232
Baudrate	600–19200
Data bits	8
Stop bits	1
Parity	None
Remote node	1, can be changed. For further information we refer to the documentation from Allen Bradley DF1 node number of the PLC system.
This node	0, can be changed. For further information we refer to the documentation from Allen Bradley DF1 node number of the terminal.

Tab. 20-1: *Port parameter settings*

20.2 Communication protocol

The Allen Bradley communication protocol DF1 is used. The communication is full duplex. The terminal sends commands to the PLC system but does not answer any commands from the PLC system.

Terminal settings

- Port parameters, see above.

Settings in the PLC system

- CRC-16 check sum
- Full duplex

20.3 Addressing

The Allen Bradley PLC system consists of a number of data files identified by a letter (identifier) and a file number. File numbers 0 to 7 are the default files created by the PLC. Additional storage may be appended by specifying the appropriate identifier and a file number from 9 to 255. Identifier B (bit), T (Timer), C (Counter), R (Control) and N (Integer) can be defined.

File type	Identifier	File number
Output	O	0
Input	I	1
Status	S	2
Bit	B	3
Timer	T	4
Counter	C	5
Control	R	6
Integer	N	7

Tab. 20-2: *Default files*

File type	Identifier	File number
Bit	B	9–255
Timer	T	
Counter	C	
Control	D	
Integer	N	

Tab. 20-3: *Additional files*

20.3.1 Addressing example

The general format in addressing

N7:15

N=file type, 7=file number, :=element delimiter, 15=element

Addressing words in an element

T4:15.1

T=file type, 4=file number, :=element delimiter, 15=element, .=word delimiter, 1=word

Addressing bits in an element

T4:15/1

T=file type, 4=file number, :=element delimiter, 15=element, /=bit delimiter, 1=bit

Addressing combinations of words and bits

I1:0.1/5

I=file type, 1=file number, :=element delimiter, 0=element number, .=word delimiter, 1=word, /=bit delimiter, 5=bit

For further information we refer to the manual for the appropriate PLC system.

20.3.2 Syntax for addressing

Inputs

Inputs are addressed as elements in the input file independent of in which slot the input unit is placed. The element number is the same as the word number in the file. Bit is the same as the bit number in the element. Word is not used.

Example ▽

PLC SLC 500 with a 2-slot expansion unit equipped with a 16 output card in the first slot and an 8 input card in the second slot. In this case the input file will look like this:
I0: 0000000000 element 0
I2: 00000000 element 1
Slot one is omitted in the input file. The address for input bit 4 in slot 2 will be I:1/4, element 1, bit 4.

△

Outputs

Outputs are addressed as elements in the output file independent of in which slot the output unit is placed. The element number is the same as the word number in the file. Bit is the bit number in the element. Word is not used.

Example ▽

PLC SLC 500 with a 2-slot expansion unit equipped with an 8 input card in the first slot and a 16 output card in the second slot. In this case the input file will look like this:
O0: 00000000 element 0
O2: 0000000000000000 element 1
Slot one is omitted in the input file. The address for ouput bit 12 in slot 2 will be O:1/12, element 1, bit 12.

△

System and integer

The element number for these files is the same as the word number in the file. Bit addressing is allowed. Word parameter is not allowed.

Binary

In this file the element number is the same as the word number in the file. Bit number must always be included. It is possible to write a greater value than 15 as the bit if the element number is omitted. In this case it is possible to address a bit direct. For example B3/100. Bit addressing is allowed. Word parameter is not allowed.

Timers, Counters and Control

An element in these files consists of three words. Valid syntax according to addressing examples 2 and 4. Predefined mnemomics may be used instead of numbers on words and bits. For further information about mnemonics we refer to the Allen Bradley manuals.

20.4 Connection to PLC

The connection is of the type “point-to-point” master/slave. The PLC system is connected to one of the communication ports of the terminal via a cable (see specification below) or modem.

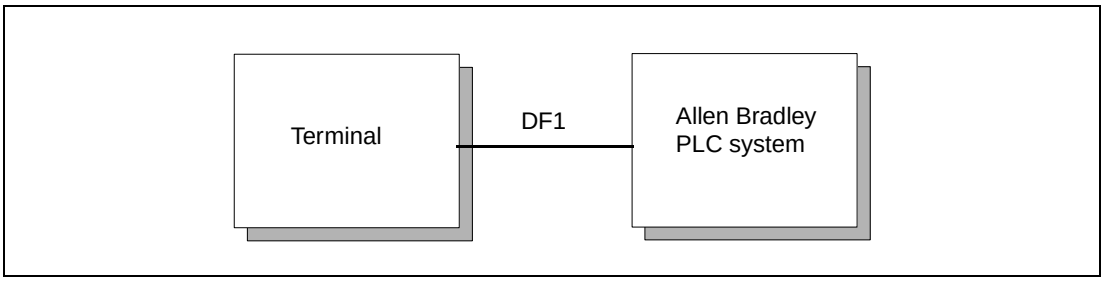


Fig. 20-1: *Connection to Allen Bradley PLC system*

NOTE Network communication via IFC 50E is not possible with Allen Bradley PLC system.

20.4.1 MicroLogix

Connect the Allen Bradley 1761-CBL-PM02 cable to the PLC system.
A special cable must be connected between the cable and the terminal.
The special cable looks like this.

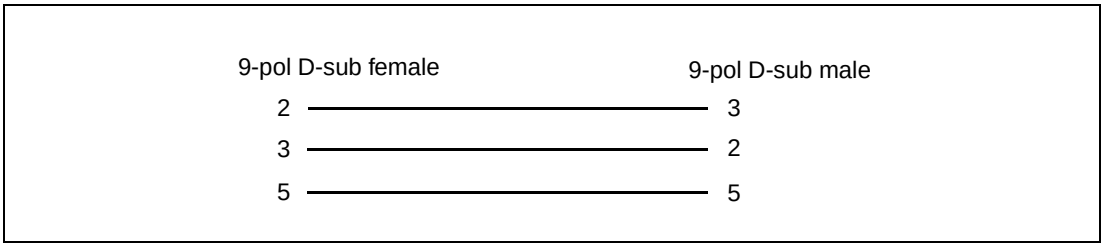


Fig. 20-2: *Wiring of the connection cable*

To use transparent mode, the PC used for programming must be connected to the 25 pol connector on the terminal via a RS422/RS485 modem.

20.4.2 RS232-C/DH485 interface, 1747-KE (SLC series)

The 1747-KE module is neccessary when using a SLC 500, SLC 5/01 or SLC 5/02 processor. There is a DF1 connector on the CPU units SLC 5/03 and SLC 5/4, but it is recommended to use the 1747-KE module. The 1747-C13 cable from Allen Bradley is connected between the DH485 connector on the PLC system and the DH485 connector on the 1747-KE module.

The standard cable MAC-PC-CAB-R2 or MAC-PROG/9-CAB is connected between the RS232 connector on the terminal and the DF1 connector on the 1747-KE module.

NOTE Transparent mode is not possible when this module is used.

21 Connection to a Siemens PLC system

This chapter describes how to connect a SIMATIC S5/S7 PLC system to the terminals in the E series and how they communicate. Addressing of an item in the SIMATIC S5/S7 PLC system is done in the normal Siemens way. For information about Siemens SIMATIC S5/S7 systems we refer to the documentation for the current system.

21.1 Connection to a SIMATIC S5

The terminal supports all systems in the SIMATIC S5 series.

The terminal can be connected to SIMATIC S5 according to the following:

- Directly to the CPU port via the protocol PG/AS511.
- Profibus, see the manual IFC PBDP.

21.1.1 Settings in the terminal

For communication with SIMATIC S5 PLC systems the following settings must be made in the terminal.

Project settings

When the project is defined you select which PLC system the terminal should be connected to. Select S5 if you will use a SIMATIC S5 PLC system.

Port parameters

Parameter	Settings
Port	RS422/RS232
Baudrate	9600
Data bits	8
Stop bits	2
Parity	Even

Tab. 21-1: *Port parameter*

21.1.2 Addressing

The terminal can handle the following data types in SIMATIC S5:

Data type English	Data type German	Description
F	M	Flag
Q	A	Output
I	E	Input
DB	DB	Datablock

Tab. 21-2: *Data types that can be handled*

The DB (Data block) in SIMATIC S5 can have a maximum length of 256 words. The terminal can access all DBs in the PLC system.

For information about instructions in SIMATIC S5, refer to the SIMATIC manual.

21.1.3 Digital signals

For digital signals, you state current bit in the byte. For example, I50.3 means bit 3 in input byte 50.

Data type English	Data type German
Ixxxxx.b	**E**xxxxx.b
Qxxxxx.b	**A**xxxxx.b
Fxxxxx.b	**M**xxxxx.b

Tab. 21-3: *Data type*

xxxxx=address 0–65535, b=bit number 0–7

21.1.4 Analog signals

For 16-bit numbers, you state the suffix W after the data type; e.g. MW100 means 2 bytes from memory byte 100-101.

Data type English	Data type German
IWxxxxx	**EW**xxxxx
QWxxxxx	**AW**xxxxx
FWxxxxx	**MW**xxxxx
DBno.**DW**adr	**DB**no.**DW**adr

Tab. 21-4: *Data type*

xxxxx=address 0–65535, no=database number 0–255 and adr=data word within the data base 0–255.

21.1.5 Wiring for the Connection to a SIMATIC S5

The connection is of the type “point-to-point” master/slave. The PLC system is connected to one of the communication ports of the terminal via a cable (see specification below) or modem.

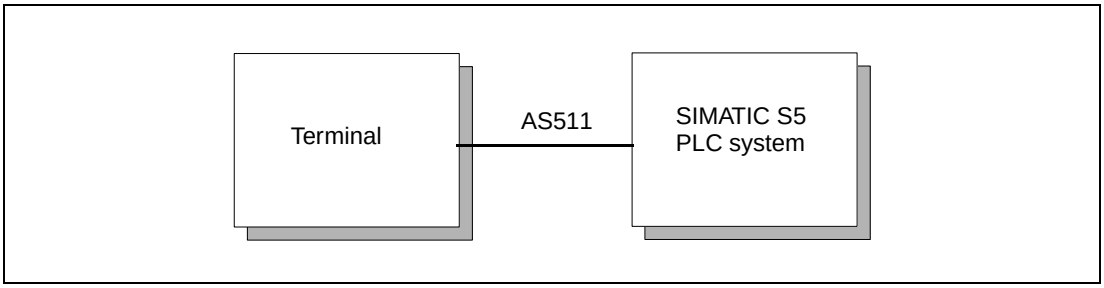


Fig. 21-1: *Connection to SIMATIC S5 PLC system*

Connect the Siemens Converter 6ES5 734-1BD20 cable to the PLC system.
A special cable must be connected between the cable and the terminal.
The special cable looks like this.

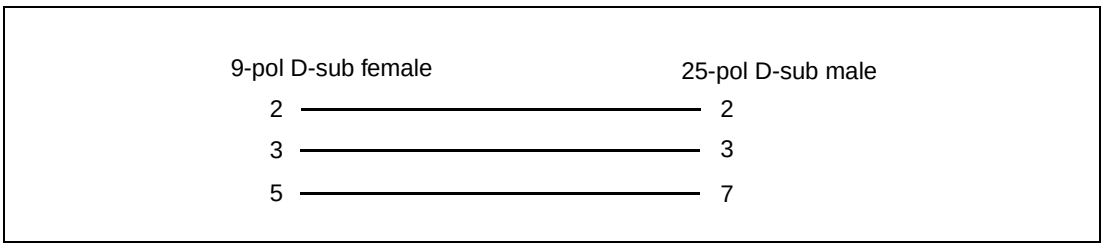


Fig. 21-2: *Wiring of the connection cable*

NOTE The function transparent mode is not available in communication with SIMATIC S5.

21.2 Connection to a SIMATIC S7

21.2.1 I/O handling for SIMATIC S7

The terminal can handle the following data types in SIMATIC S7:

Data type English	Data type German	Description
F	M	Flag
Q	A	Output
I	E	Input
DB	DB	Datablock

Tab. 21-5: *Data types that can be handled*

The project memory decides the max length of the DB (Datablock) in SIMATIC S7. The terminal can access one DB in the PLC system.

All data types consists of byte areas. Addressing is always byte-specific, regardless of whether it is 1, 16 or 32 bits. The addresses are always decimal, 0–65535.

For information about the instructions in S7 we refer to the SIMATIC manual.

Digital signals

For digital signals, you state current bit in the byte. For example I50.3 means bit 3 in input byte 50.

Data type IEC	Data type SIMATIC
Ixxxxx.b	**E**xxxxx.b
Qxxxxx.b	**A**xxxxx.b
Mxxxxx.b	**M**xxxxx.b

Tab. 21-6: *Data type*

xxxxx=address 0–65535, b=bit number 0–7

Analog signals

For 16-bit numbers, you state the suffix W after the data type; e.g. MW100 means 2 bytes from memory byte 100–101.

Data type IEC	Data type SIMATIC
IWxxxxx	**EW**xxxxx
QWxxxxx	**AW**xxxxx
MWxxxxx	**MW**xxxxx
DBWxxxxx	**DBW**xxxxx

Tab. 21-7: *Data type*

xxxxx=address 0–65535.

22 Appendix

22.1 Erasing the memory

In the menu Setup in the terminal you will find the function Erase memory. The funtion erase the terminal's application memory. All blocks and definitions of alarms, time channels, function keys and system signals are erased.

Parameter	Description
Enter	The memory is erased. The configuration menu is shown automatically when the erasure is completed.
Prev	Return to the previous level without erasing the memory.

Tab. 22-1: *Erasing the memory*

NOTE

All data stored in the terminal is lost when the memory is erased. The language parameter are not affected by this function. Other parameters are erased or allocated the default setting values.

22.1.1 Down loading the system program

The terminal has a system program (operative system) which is stored in the terminal's memory when delivered. The system program can be exchanged, for example when updating to newer versions. The following is required to transfer the system program to the terminal:

- Personal computer
- Transfer cable between thePC software and the E900
- The SYSLOAD.EXE personal computer program (available as an icon in the program group HMI Tools).
- File with the new system program (e.g. E900V100.BIN)

Transfer in the following steps:

① Connect the cable between the personal computer and the terminal.

② In E900 you hold the [←] and [F1] keys pressed on start up. A LED should then blink green and the screen should be off.

③ The personal computer program is started when you double click on the icon Sysload in the program group HMI Tools.

No settings need to be made on the terminal.

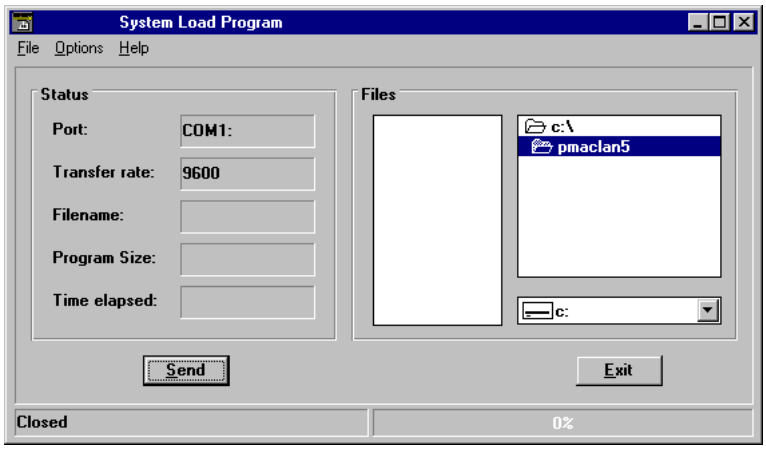


Fig. 22-1: *System Load Program page*

Under **Options**, **Settings** you set the communication port and transfer speed for the communication.

Down loading of the system program can also take place through a modem for remote updating of terminals. Then the baudrate must be 2400 baud.

22.1.2 Information page

The terminal contains an information page. In E900 the information page is activated by simultaneously pressing the [←] and [PREV] keys in run mode.

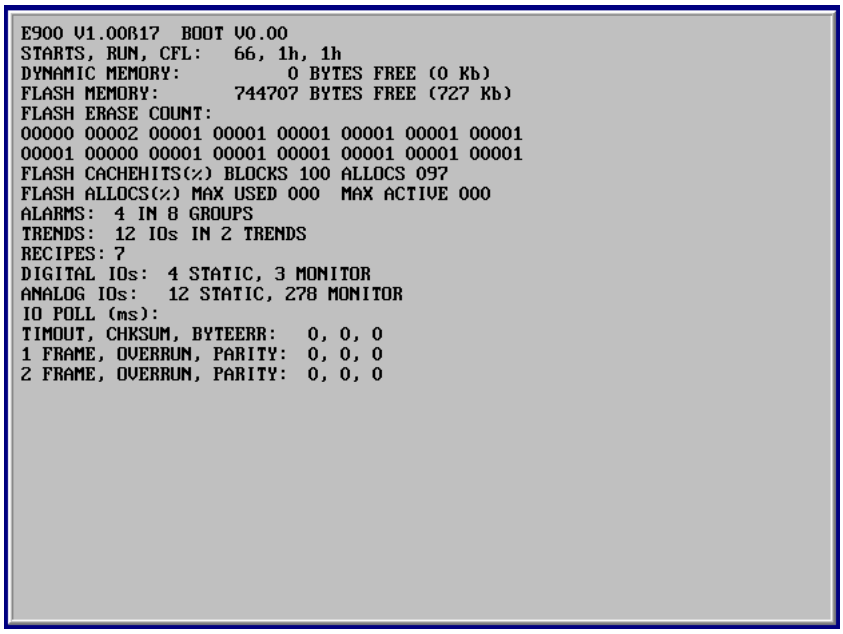


Fig. 22-2: *Information page*

Parameter	Description
STARTS	Number of times the terminal has been started.
RUN	Number of hours the terminal has been in operation.
CFL	Number of hours the backlighting has been switched on.
DYNAMIC MEMORY	Free RAM memory (working memory) in number of bytes.
FLASH MEMORY	Free Flash memory (project memory) in number of bytes.
FLASH ERASE CNT	Number of times each sector of the Flash memory has been erased. The current spareblock always shows zero.
FLASH CACHEHITS	Percentage of Block/Allocation cachehits in the filesystem.
FLASH ALLOCS	Max percentage of used/active allocations per block in the file system.
ALARMS	Number of alarms and alarm groups in the application.
TRENDS	Number of trend IO and trend objects in the application.
RECIPES	Number of production methods in the application.
DIGITAL IOs	Number of digital IOs monitored continuously (static) and number in the current block (monitor).
ANALOG IOs	Number of analog IOs monitored continuously (static) and number in the current block (monitor)
IO POLL	Time in ms between two readings of the same IO.
TIMEOUT	Number of time-outs in the communication.
CHKSUM	Number of check sum errors in the communication.
BYTEERR	Number of byte errors in the communication.
FRAME	Number of framing errors in each port.
OVERRUN	Number of overrun errors in each port.
PARITY	Number of parity errors in each port. 1=RS422 port and 2=RS232 port.

Tab. 22-2: *Parameters of information page*

22.1.3 Project memory

The project memory in the terminal is 2 Mbytes. These bytes can be freely assigned to different functions. Information about free memory available is presented on the information page.

23 Glossary

Alarm group

Alarms can be sorted into groups; e.g. by danger levels.

Alarm handling

Function for detection and presentation of alarms.

Alarm list

A list presenting active, acknowledged and unacknowledged alarms.

Alarm message

Freely defined text linked to a signal that is displayed in the alarm list.

Block

Screens in the terminal are called blocks. A project is composed of blocks arranged hierarchically or sequentially.

Block head

Contains the basic definitions of a block.

Block number

The number of a block in the terminal. Can be between 0 and 999.

Block type

There are two types of blocks, graphic and text. Graphic blocks are used to create operator dialogs. Text blocks are used to create reports etc.

Communication port

Communication connections.

Configuration mode

All basic settings for the terminal is made in the configuration mode.

Control block

Number of data registers in the PLC system controlling the communication.

Data type

Type of signal in the PLC system, e.g. input, output, register and memory cell.

Dynamic object

Objects connected to signals in the PLC system.

Fieldbus

Network for vendor-independent communication.

Function key

Key connected to a signal in the PLC system. The signal is affected when the key is pressed.

Graphic block

Blocks created by graphic objects.

Historic trending

Analog values from the PLC system are stored in the terminal and presented in a trend object during run mode.

MAC Programmer+/SW-MTA-WIN

Programming tool for a PC with Windows. The terminals are programmed via the PC software.

Manoeuvring

Manoeuvring means that objects can be affected in run mode.

Master

The master in the network controls the communication.

Menu tree

A project can be built up as a menu tree with one block as an overview and blocks underneath containing detailed information.

Network

Several connected units which can communicate with each other.

No protocol mode

For communication with external units e.g. bar code readers.

Object

A project is built up with objects. There are predefined objects such as VU meter, bar graph, trend object and diagram object.

Object oriented work methods

Everything in the terminal is based on objects. The user selects the objects function. All signals are defined this way.

Operator screen

A screen in the terminal from which the operator supervises/maneuvers the application. A project normally contains of several operator screens, blocks.

Password

The operator must use a password, to gain access to a block or object. The password can be connected to blocks and manoeuvrable objects.

PROFIBUS

Vendor-independent communication protocol for communication with PLC systems.

Programming mode

The terminal is programmed in the programming mode.

Project

One project is built up with blocks.

Real time clock

The terminal contains a real time clock.

Recipe handling

Function for storing and re-using parameter settings.

Report printout

Via textblocks printouts can be generated; for example, production follow up.

RS232

Communication interface.

RS485

Communication interface.

Run mode

The mode in which the project is running with the PLC system.

Scaling

Scaling of register values to engineering units.

Security level

Block and manoeuvrable object can be assigned a security level. This means that the user must enter a password to be able to maneuver the object.

Sequence control

A project can be built up as a sequence where the blocks are displayed in a predefined order. Change of blocks is normally controlled by the PLC system.

Signal format

A suffix added to the signal presenting double register, rounded-off values and/or only positive values.

Slave

A communication network consists of a master and one or more slaves. The slave follows instructions from the master.

Static graphic object

Objects used to draw static graphic, lines, circles and rectangles etc.

Static text

Text in the block that will not be changed during program execution.

Status word

Affect the look of the display in run-time mode.

Status

The value of the signal in the PLC system is shown on the display.

Symbols

Pictures in BMP format can be created, imported and exported.

System block

Reserved blocks in the terminal used for the alarm list, time channels and function keys etc.

Time control

Possibility to control signals related to the real-time clock.

Transparent mode

Communication method. Means that, for example, the PLC system can be programmed from a PC with the programming tool at the same time as the terminal is connected to the PLC system.

INDEX

J

K

L

M

N

O

P

R

S

T

V

W