E-Designer

Programming Software for Text Operator Terminals E50, E100, E150 and E200

Operation Manual

⮯ **MITSUBISHI ELECTRIC** INDUSTRIAL AUTOMATION

About this Manual

The texts, illustrations, diagrams and examples in this manual are only intended as aids to help explain the functioning, operation, use and programming of the E-Designer programming and documentation system.

If you have any questions regarding the installation and operation of the software described in this manual, please do not hesitate to contact your sales office or one of your Mitsubishi distribution partners.

You can also obtain information and answers to frequently asked questions from our Mitsubishi website under

www.mitsubishi-automation.com.

The E-Designer software is supplied under a legal license agreement and may only be used and copied subject to the terms of this License Agreement.

October 2004

Safety instructions

General

Read the safety precautions carefully.

Check the delivery for transportation damage. If damage is found, notify the supplier as soon as possible.

The terminal fulfills the requirements of article 4 of EMC directive 89/336/EEC.

Do not use the terminal in an environment with high explosive hazards.

The supplier is not responsible for modified, altered or reconstructed equipment.

Use only parts and accessories manufactured according to specifications of the supplier.

Read the installation and operating instructions carefully before installing, using or repairing the terminal.

Replacing the battery incorrectly may result in explosion. Only use batteries recommended by the supplier.

Never pour fluids into any openings in the terminal. This may cause fire or electrical shock.

Only qualified personnel may operate the terminal.

During installation

The terminal is designed for stationary installation on a plane surface.

Put the terminal on a stable surface during installation. Dropping it or letting it fall may cause damage.

Install the terminal according to the accompanying installation instructions.

Ground the terminal according to the accompanying installation instructions.

Only qualified personnel may install the terminal.

Separate the high voltage, signal and supply cables.

Make sure that the voltage and polarity of the power source is correct before connecting the terminal to the power outlet.

The openings on the enclosure are for air convection. Do not cover these openings.

Do not place the terminal where it might be exposed to strong magnetic fields.

Do not install the terminal in direct sunlight.

Peripheral equipment must be appropriate for the application.

Some terminal models have a laminated film over the display to reduce the risk of scratches. To avoid static electricity that might damage the terminal, carefully remove the film.

UL Installation

Power, input and output (I/O) wiring must be in accordance with Class I, Division 2 wiring methods (Article 501-4 (b) of the National Electrical Code, NFPA 70) and in accordance with the authority having jurisdiction.

During use

Keep the terminal clean.

Emergency stop and other safety functions may not be controlled from the terminal.

Do not touch the keys, displays, etc. with sharp objects.

Be aware that the terminal is operable and registers button presses and input via the touch screen even when background lighting is not on.

Service and maintenance

The agreed warranty applies.

Clean the display and face with a soft cloth and mild detergent.

Only qualified personnel should carry out repairs.

Dismantling and scrapping

The terminal or parts thereof shall be recycled according to local regulations.

The following components contain substances that might be hazardous to health and the environment: lithium battery, electrolytic capacitor and display.

6   Text-based presentation and maneuvring

7   Message library

8   Alarm handling

9   Recipe handling

20   Technical data

1 Introduction

Production rates in modern industrial environments are steadily increasing and operator tasks at machines or on production lines are becoming increasingly more complex and involve more responsibility. The operator must be able to obtain information on current status easily and quickly, and be able to influence the operation of the machine just as easily. The functions in the controllers are also increasing and becoming more advanced, enabling more complicated processes to be controlled efficiently. The operator terminals make human-machine communication simple and safe even for the most advanced processes.

The graphical operator panels have been developed to meet the requirements set for human-machine communication when controlling or monitoring different applications in the manufacturing and process industries, etc. They simplify the operator's work since they can easily be adapted to the working environment, which means that the operator can continue to use the concepts he or she is familiar with.

Projects in the terminal can be built up as menu hierarchies or as sequences. A menu hierarchy consists of a main menu (with, for example, an overview) and a number of underlying images with more detailed information on special sections. The operator normally selects which image is to be shown.

The images in the operator terminals are called blocks
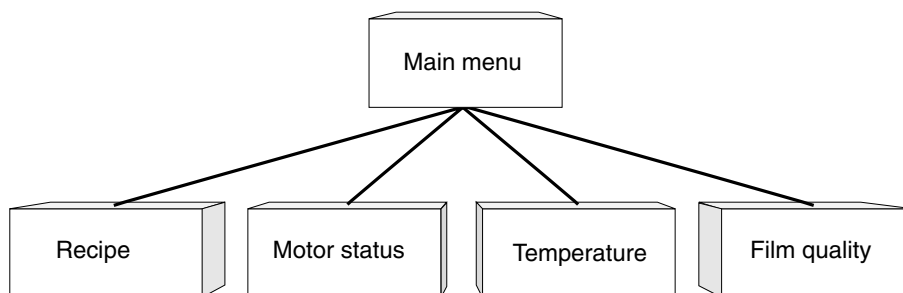
A sequence is also based on a main menu, from which the operator selects a sequence showing the blocks in a predetermined order. Normally displaying of the blocks is controlled by the controller program.

```
                    ┌──────────────┐
                    │  Main menu   │
                    └──────────────┘
                   ╱                ╲
          ┌──────────────┐   ┌──────────────┐
          │  Pre-wash    │   │    Setup     │
          └──────────────┘   └──────────────┘
                 │
          ┌──────────────┐
          │  Main wash   │
          └──────────────┘
                 │
          ┌──────────────┐
          │   Rinsing    │
          └──────────────┘
                 │
          ┌──────────────┐
          │     Spin     │
          │   drying     │
          └──────────────┘
                 │
          ┌──────────────┐
          │    End of    │
          │   program    │
          └──────────────┘
```
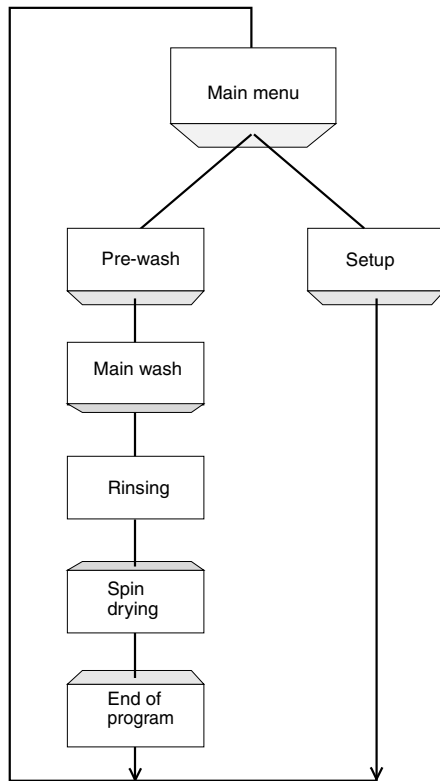
The functions in the terminal enable text-based presentation and maneuvring. There are also functions for alarm handling, printouts, recipe handling, and time control, etc.

The functions are not only easy to use in the terminal, they are also cost-efficient in comparison with conventional solutions with buttons, indicator lamps, time relays, preset counters and seven-day clocks. There are also functions to increase the application of the controller.

MITSUBISHI ELECTRIC

1.1   Programming

The terminal is programmed through a PC with the programming tool. It is also possible to program the terminal through the built-in keyboard.

The terminal is to a large extent object oriented, i.e. first an object is selected and then the function the object is to have. All types of signals are defined on this principle.

The programmed project is stored in the terminal.

1.2   Connection of the terminal to the controller

There are many advantages in using a terminal together with the controller. The user does not need to make any changes or additions in the existing controller programs and the terminal does not block any of the inputs or outputs in the controller. The overview of functions such as time control and alarm handling in the controller is also improvedt.
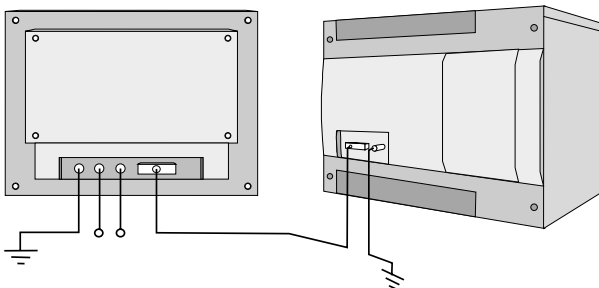
1.3   Status display and control

The operator is familiar with indicator lamps and analog and digital display instruments since these are used in the majority of applications today. The same applies to push buttons and rotary and thumbwheel switches for controlling a system. The terminal enables the operator to have all status displays and controls in one unit.

The operator can easily see and influence information in the controller. Moreover, it is possible to clearly see and influence all the signals affecting a specific object, which further simplifies the work.
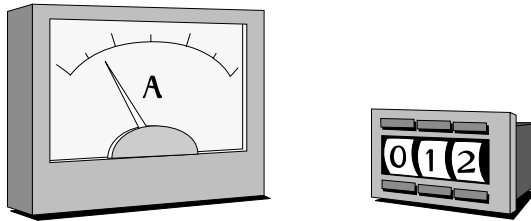
This is possible thanks to the fact that the interchange of all information takes place through the so-called blocks in the terminal. Blocks can be of the text block type, with only text information.

The terminals have function keys for direct maneuvring. Maneuvring is controlled by linking different commands to the function keys.

If several blocks are used, the operator can jump between the different blocks by using jump objects. This creates a menu hierarchy, which produces a structured application.

🔶 **MITSUBISHI ELECTRIC**

1.4　　Location of the terminal

The terminal should be placed at the workplace to ensure maximum usability. This will enable the operator to receive information directly and to respond quickly and efficiently.

The terminal should be positioned at the correct working height so that the operator can easily see and maneuver the terminal. Screen visibility is influenced by distance, height, angle, sunlight and the selection of color.

Operational monitoring, maneuvering and maintenance can also be performed remotely from another location in the building or from another site, for example. Communications can then be, for example, via the internal network, the Internet or modem. If the production line is long with many workplaces, several terminals can be connected to one or more control systems in the network.
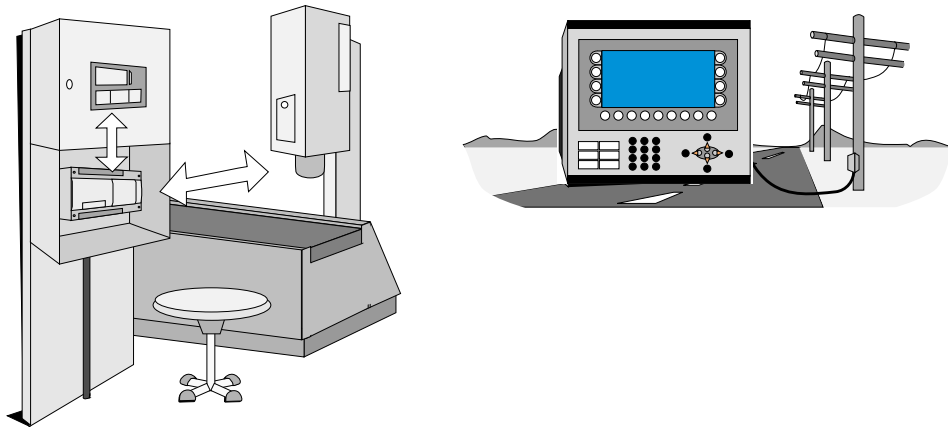
1.5　　Compact solutions

External units such as barcode scanners, weighing machines, and modems, can be connected through the terminal to the controller. All that is required is for the unit to be connected to an RS232 interface, and that communication is made through ASCII protocol. Data entering the terminal is written directly to the controller register.

A unit working in parallel may also be connected. It can be an additional terminal or a personal computer with the programming tool for the control system. Via the terminal, the control system can be programmed at the same time as the terminal is communicating with the control system.

When two remote systems are to connected to one and the same terminal (dual drivers are used in the terminal), it is possible to exchange data (analog and digital signals) between the control systems.

For information on installation, please see the installation manual that was supplied with the ter-

minal.

Operator terminal with control system and barcode reader connected.

Dual drivers can be used when different systems are connected.
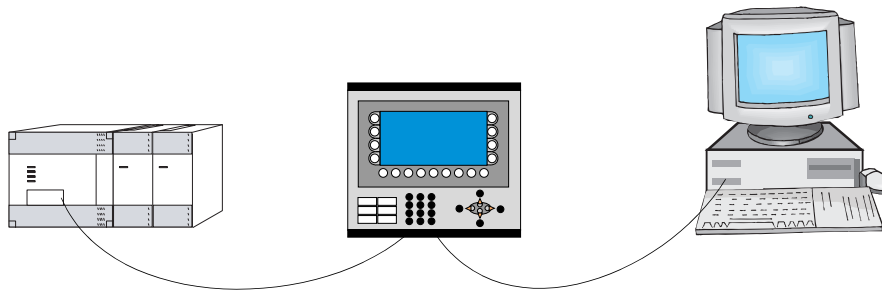
Operator terminal in Transparent mode.

MITSUBISHI ELECTRIC

2   Installation

2.1   The programming tool

The programming tool is a program package used to develop projects for operator terminals in the E-series. The functions in the programming tool depend on which terminal is used.

In the programming tool we recommend using a mouse. Refer to the Windows User's Guide for short commands.

In the programming tool a project is created with text blocks, which are then transferred to the terminal. The programming is described in chapter *Programming with the programming tool.*

Help texts are available for all functions. The help text for the current function is obtained by pressing the F1 key. By pressing the help button in the toolbox and then clicking on a function, information is shown on the function.

2.1.1   System requirements

To use the programming tool a PC with at least 55 Mb of available memory and Microsoft Windows 9x/NT/2000/Me/XP is required. The programming tool can be used on either a color or monochrome screen. Microsoft Internet Explorer version 5.0 or later must be installed on the computer.

2.1.2   Installing the programming tool

The programming tool is supplied on a CD. When you place the CD in your CD ROM drive the installation will start automatically. If not, select **RUN** in the **Start** menu and enter D:\setup.exe (if D: is the CD ROM drive). Select to install the programming tool by clicking on the name and following the instructions.

The installation creates an icon for the programming tool in the E-Designer group. You can find the programming tool by clicking on **Start** and selecting **Program/E-Designer/E-Designer**. The manual can be read directly from the CD by clicking on **Manuals**.

2.1.3   The menu bar

From the menu bar you can reach a number of pull-down menus; **File**, **Edit**, **View**, **Functions**, **Setup**, **Object**, **Layout**, **Transfer**, **Window** and **Help**. The file menu contains functions affecting the entire project. In the Edit menu there are functions to cut, copy and paste. The different editors, e.g. **block managers**, **alarm manager** and **symbol manager** are shown in the **View** menu. The basic settings for the terminal are made in the **Setup** menu. The **Object** menu is only available in the different managers and contains all the objects. The objects are also included in the toolbox.

2.1.4   The status field

The status bar is shown at the bottom of the window in the programming tool.On the **View** menu there is a function for showing/hiding the status field.

The left part of the status field describes the menu function for the function selected in the menu. A short description of the function the cursor points to is shown for the functions in the toolbox.

The right part of the status field indicates which of the following keys are activated:

OVR   Overwrite (Insert key)
CAP   Caps Lock
NUM   Num Lock

Coordinates, line and column, in the block manager are also shown.

245   148 SCRL CAP NUM

2.2   Connecting the terminal to a PC
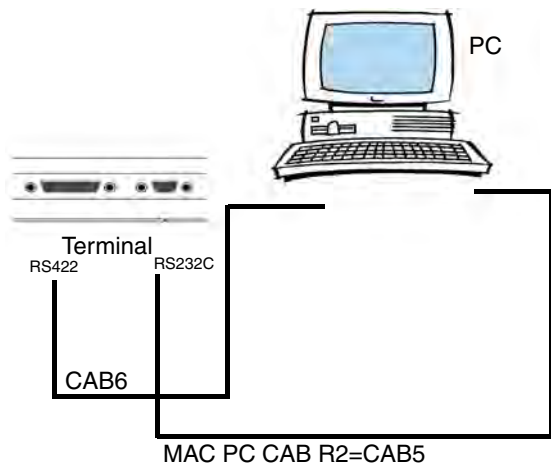
To connect the controller to the terminal, use the cable recommended in the driver manual for the selected controller.

NOTE   ▮ The power must be switched off when connecting.

▲ MITSUBISHI ELECTRIC

2.3   Downloading the system program (with HMI Tools)

Not applicable for E50.

In the terminal there is a system program (operative system), which is stored in the terminal's memory on delivery. The system program can be exchanged, e.g. to update to newer versions. Transferring the system program to the terminal requires the following:

- PC

- Transfer cable between the PC and the terminal.

- The PC program System Loader(available as an icon in the program group HMI Tools). See the manual for HMI Tools to install the program group.

- File with new system program (*.bin).

NOTE   ∎ All communication between the PC and the operator terminal E50 is conducted through the programming tool. The system program is transferred at the same time as the driver. See the chapter *Transferring projects.* The PC programs in the program group HMI Tools cannot be used together with E50. The cable CAB6 cannot be used to transfer projects to E50. The adapter 232-25-ADP must be used on the RS422 port perform a transfer with a CAB5 cable. The adapter is supplied with E50.

Transfers are made in the following steps:

① Connect the cable between the PC and terminal.

② The PC program is started by selecting Program\HMI Tools\System Loader in the **Start** menu.

No settings need to be made in the terminal.
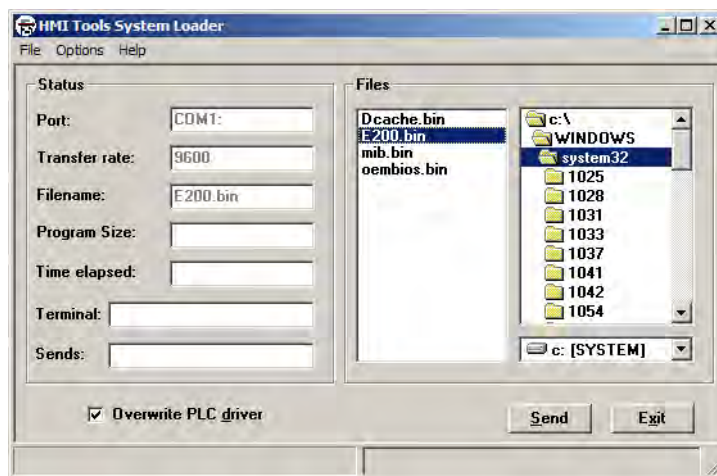
Under **Options/Comm Settings** you can set the communication port and transfer speed.

The system program can also be downloaded through a modem for remote updating of terminals.

NOTE   ∎ When downgrading a terminal it is important to ensure that the box **Overwrite PLC driver** is marked.

If downloading of a new system program (*.bin file) fails when you click **Send** in the HMI Tools System Loader, the terminal is automatically placed in sysload mode. When the terminal is in sysload mode, it is ready to make a new try at downloading the system program.

MITSUBISHI ELECTRIC

3   Terminal functions

This chapter describes the different modes in the terminal, the keyboard and the information page in the terminal. For a description of the different key sequences, see chapter *Key sequences.*

3.1   The operation modes, RUN, PROG and SETUP

The terminal has three operation modes. E50 only has run mode and transfer mode.

● The configuration mode, Setup, where the basic settings are made, e.g. selection of controller, menu language and printer settings.

● Programming mode, Prog, where the terminal can be programmed and where changes can easily be made during system set-up.

● Run-time mode, Run, where the application is run.

3.1.1   Switching between modes

Switch between PROG and RUN

Switch between programming, PROG, and run-time, RUN, by pressing the keys [←] and [MAIN] simultaneously. The possibility of switching between PROG and RUN from the terminal can be controlled with a password. For further information, refer to chapter *Programming with the pro-gramming tool.*

Switch between PROG and SETUP

To switch between programming, PROG and configuration, SETUP, press [LEAVE] until the start-up menu is shown and then press any key. Press [LEAVE] to return to PROG.

Switch between RUN and SETUP

Press [←] and [MAIN] simultaneously to enter programming mode, PROG. You can now press any key when the start-up menu is shown to enter the configuration mode, SETUP. To return to run mode, RUN, press [←] and [MAIN].

The following key sequences correspond to the key sequence [←] + [MAIN]:

E100/E150[←] + [ENTER]

3.1.2   Configuration mode, SETUP

This section describes functions which cannot be performed with the programming tool. For settings through the programming tool refer to chapter *Programming with the programming tool.*

Erasing the memory

In the Setup menu in the terminal there is a function **Erase memory**. This function erases the terminal's application memory. All blocks and definitions for alarms, time channels, function keys and system signals are erased.

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

NOTE   I   When the memory is erased all the data stored in the terminal is lost. The language selection parameter is not affected by this function. Other parameters are erased or given their default values.

Contrast setting

Contrast is set on the back of the terminal.

NOTE   I   Contrast is affected by ambient temperature. If the terminal is programmed in a room where the temperature deviates significantly from the location where the terminal will be placed, any contrast settings should be made after 15-30 minutes in the actual ambient temperature.

3.1.3   Programming mode, PROG

Edit

The application is built up in the programming tool. Certain modifications can, however, be made when the terminal is in the programming mode by choosing **Edit**. The help line in the terminal shows what can be done using the function keys.

Transfer

The project is normally transferred to the terminal form the programming tool. By using the function **Automatic terminal RUN/TRANSFER switching** in the programming tool, the program automatically sets the terminal in transfer mode. When the terminal is in programming mode, it can be manually set to the transfer mode by choosing **Transfer**.

♦ **MITSUBISHI ELECTRIC**

3.1.4   **Run mode, RUN**

Run mode is the mode where the application is run. Block 0 will automatically be shown on the display when transferring to run mode.

The built-in keyboard in the terminal is used to mark and change values in run mode.

If a communication error occurs between the terminal and the controller an error message will be shown on the display. The terminal starts automatically when the communication is resumed.

To obtain a watch-dog function where the controller can detect if a communication error has occurred, the terminal clock can be continuously transferred to a register in the controller (see section Date/Time format is chapter *Programming with the programming tool)*. The controller checks if the register is updated, and if not an alarm indicating a communication error is activated in the controller.

The descriptions of objects/functions also include how the different objects/functions function in run mode.

3.2   The keyboard in the terminal
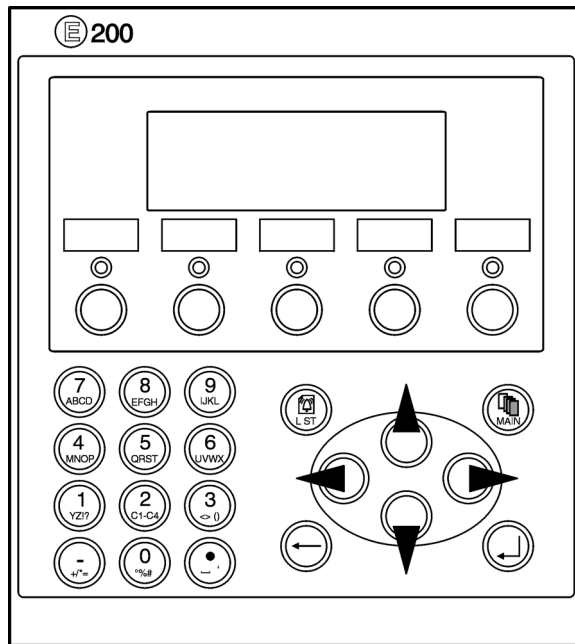
3.2.1   Alphanumeric keys

From the alphanumeric keyboard the following characters can be entered in dynamic text and numerical objects during run mode in the terminal.

0-9
A-Z
a-z
! ? < > ( ) + / * = ° % # : ' @
National characters

NOTE   ▮ No characters can be entered from the keyboard on E50 because it does not have alphanumeric keys.

Numeric values are entered by pressing the respective key once. A to Z are entered by pressing the respective keys two to five times, and the lower case letters a to z are entered by pressing the respective keys 6 to 9 times. The delay time interval between pressing can be set (see section, Terminal options in chapter *Programming with the programming tool*). If the key is not pressed within the delay time interval the cursor moves to the next position.

National characters are entered by pressing the keys C1-C4 two to nine times. This enables non-standard characters on the alphanumeric keyboard to be available in the terminal. (See section, Country settings in chapter *Programming with the programming tool*).

NOTE   ▮ In the programming tool all characters in the selected character set (see section, *Country Settings* in the chapter *Programming with the programming tool*) are used in static text, with the exception of reserved characters. Enter the required character by pressing the Alt key and simultaneously pressing 0 (zero) on the numeric keyboard on the PC, followed by the character code.

⯈ MITSUBISHI ELECTRIC

Numeric and alphanumeric keys in E100

To enter numbers in E100 the NUM key is held pressed while the required number is pressed. Alternatively the command NTGL on the command line under **System signals** in the **Setup** menu is given. The command NTGL enables the NUM key to have a toggle function. This means that numeric input is activated when you press the NUM key. The keys now correspond to numbers. Deactivate the input of numeric characters by pressing the NUM key.

To enter letters, in for example a text object in E100, open the object for editing by pressing Enter. Now do as follows:

① Press the NUM key + Enter key simultaneously.

② Select characters in the options list by using the up and down arrow keys.

③ Press the Enter key.

④ Select new character, etc.

⑤ The position of the cursor is moved with the left and right arrow keys. The character. (dot) is obtained by pressing the 0 key twice.

Reserved characters

The ASCII characters 0-32 (Hex 0-1F) and 127 are reserved for internal functions in the terminal and must be used in any projects or files in the terminal. The characters are used as control characters.

3.2.2   Arrow keys

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

3.2.3   Built-in function keys

Not all the keys are available on all terminals.

The key [↵] ([ENTER]) is used to confirm the setting made and to go to the next line/level.

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

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

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

The key [MAIN] is used to jump to block 0 in run mode.

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

NOTE   ▮ When the main block (block number 0) has been shown the [PREV] key will not work, since the block history is deleted when the main block is shown.

3.2.4   Key sequences

The terminal has key sequences for the following functions.

Terminal-model	Function			
	Sysload	**Self-test**	**Switch between PROG and RUN**	**Diagnostic page**
E100	[←] + [F1]	◀ + ▶	[←] + [ENTER]	[←] + ▶
E150	[←] + [F1]	◀ + ▶	[←] + [ENTER]	[←] + ▶
E200	[←] + [F1]	◀ + ▶	[←] + [MAIN]	[←] + [LIST]

E50 only has run mode and transfer mode. The following key sequences are applicable for E50.

Key sequence	Function
[F3] [F4]	Sets the terminal in manual transfer mode, Transfer.
◀ ▶	Activates the self-test function.

3.3   Setting the real-time clock

Not applicable for E50.

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

Select the alternative **Set Terminal clock**. The date and time are now shown. Press [SET] to change the settings. Enter the required date and time. In editing mode you move the cursor with the arrow keys. Press [NEXT] to return or cancel the setting before pressing [ENTER].

The real-time clock can also be set in run mode through a maneuvrable clock object and during the transfer of projects from a PC to the terminal, see chapter *Transferring projects*.

❚ By using a command, a digital signal can be used to indicate when the real-time clock's battery needs to be changed. See the section *Commands* in the chapter Programming with the programming tool.

▲ **MITSUBISHI ELECTRIC**

3.4   Diagnostic page

The terminal contains an diagnostic page. The diagnostic page is activated by pressing the key combination [←] and [PREV] simultaneously in RUN mode. A function or touch key can also be used/configured to call up the information page.

The current terminal, system program version and hardware version are shown at the top of the information page.

Parameter	Description
STARTS	Number of times the terminal has been started.
RUN	Number of hours the terminal has been running.
CFL	Number of hours the backlighting has been switched on.
DYNAMIC MEMORY	Available RAM memory (working memory) in number of bytes.
FLASH MEM PROJ	Available Flash memory (project memory) in number of bytes.
FLASH MEM BACK	Available Flash memory on the backup memory card (IFC MC) in number of bytes.
FLASH CACHEHITS	Percent of block/allocation cache hits in the file system.
FLASH ALLOCS	Max percent of used/active allocations per block in the file system.
DRIVER 1	Current driver and driver version.
DIGITAL I/O's	The number of digital signals linked to controller 1 continuously monitored (STATIC) and the number in the current block (MONITOR).
ANALOG I/O's	The number of analog signals linked to controller 1 continuously monitored (STATIC) and the number in the current block (MONITOR).
I/O POLL	The time in ms between two readings of the same signal in controller 1.
PKTS	The number of signals in each package transferred between the terminal and controller 1. See section *Efficient communication* in chapter General.
TOUT1	The number of timeouts in communication with controller 1.
CSUM1	The number of checksum errors in communication with controller 1.
BYER	The number of byte errors in the communication.
DRIVER 2	Current driver and driver version. The parameters for Driver 2 are only shown if controller 2 is defined in the project.
DIGITAL I/O's	The number of digital signals linked to controller 2 continuously monitored (STATIC) and the number in the current block (MONITOR).
ANALOG I/O's	The number of analog signals linked to controller 2 continuously monitored (STATIC) and the number in the current block (MONITOR).
I/O POLL	The time in ms between two readings of the same signal in controller 2.
PKTS	The number of signals in each package transferred between the terminal and controller 2. See section *Efficient communication* in chapter General.
TOUT2	The number of timeouts in communication with controller 2.
CSUM2	The number of checksum errors in communication with controller 2.
1/2/3	Current port for FRAME; OVERRUN and PARITY. 1=RS-422-port, 2=RS-232C-port and 3=RS-485-port.
FRAME	The number of frame errors in each port.
OVERRUN	The number of overrun errors in each port.
PARITY	The number of parity errors in each port.

3.5   "Joystick function

Not applicable for E50.

This function makes it possible to use the arrow keys as function keys. On the command line under **System Signals** you enter the command AK and an address. For example, AKM100 (the command AK and memory cell M100).
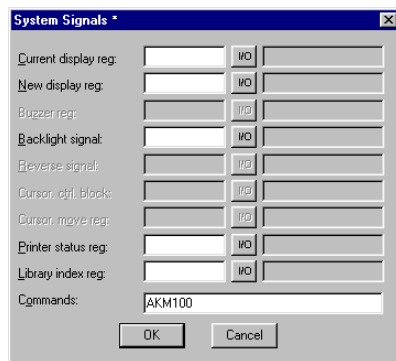
Memory cell M100 is the enable signal and the following 4 memory cells have functions according to the following control blocks.

Memory cell	Description
Mn0	Enabled. The arrow keys are disabled. Disabled = normal function.
Mn1	Left arrow
Mn2	Down arrow
Mn3	Up arrow
Mn4	Right arrow

If you press on an arrow when the enable signal is on, the memory cell corresponding to the key you press will be set to one. When the enable signal is set to one the arrow keys will not have their normal functions.

Example   The following examples can be used to disable the joystick function and return to the normal function. Do as follows:

① Create a block defined as 1st man obj.

② Enter the static text JOYSTICK.

③ Create a digital object with the following settings:

◆ **MITSUBISHI ELECTRIC**
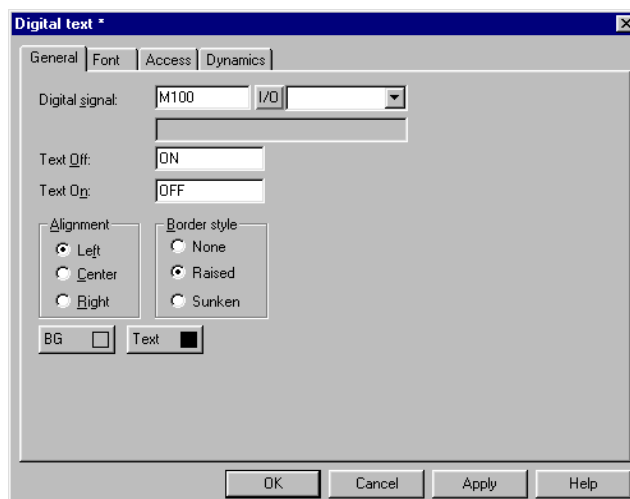
Click **Enable operator input** under the tab **Access**.
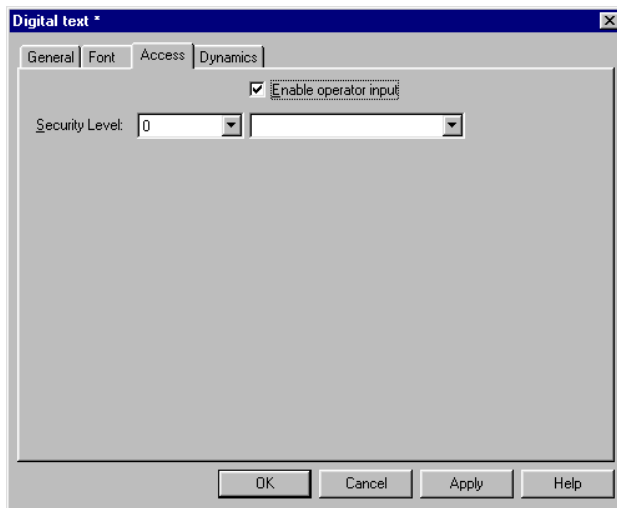
Click **OK**.

The text block after making the settings in the example

MITSUBISHI ELECTRIC

4   General

This chapter describes the system structure of the terminal and its basic functions. There is also an explanation of the general principles, object parameters, and joint functions applicable in the terminal.

4.1   Method for programming a project

The graphical structure of the application in the terminal constitutes a well-arranged monitoring tool for the operator. It is important to organize the application well and to consider which functions are necessary. Start with the overall view, and then work down to the detailed level. When a project is programmed you start with the functions in your application. Each function becomes one or more blocks, depending on how complex the function is. A project can contain text blocks, and each block can contain static and dynamic objects. The blocks should be arranged in hierarchies to achieve a structured application, and to simplify work procedures for the machine operator. The application can also be organized as sequence controls.

It is possible to test the complete application, before it is put in run mode.
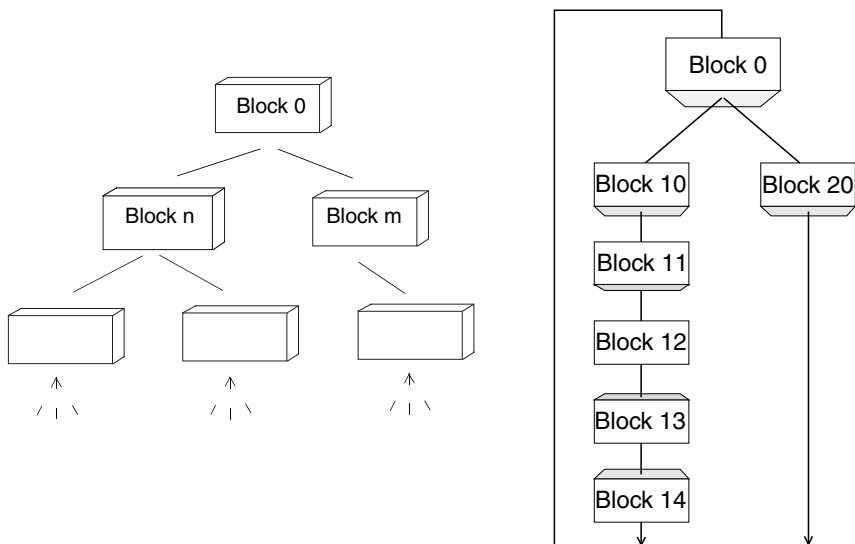
4.2   Efficient communication

To make the communication between the terminal and the controller quick and efficient the following should be noted about how the signals are read and how the reading can be optimized.

4.2.1   Signals affecting the communication time

Only signals to objects in the current block are read continuously. This also includes signals for object dynamics. Signals to objects in other blocks are not read, thus the number of blocks does not affect the communication time.

Besides the signals to objects in the current block, the terminal reads the following signals from the controller continuously:

● Display signals (Block Header)

● Print signals (Block Header)

● LED registers

● Alarm signals

● Remote acknowledge signals on alarms and alarm groups

● Login signal (Passwords)

● Logout signal (Passwords)

● Registers for trend curves

● Bargraph registers if using min/max indicators

● New display register

● Buzzer register

● Backlight signal

● Cursor control block

● Recipe control block

● Library index register

● Index Registers

● Controller clock register if the controller clock is used in the terminal

● List erase signal (Alarm Settings)

● No protocol control register

● No protocol on signal

4.2.2   Signals not affecting the communication time

The following signals do not affect the communication time:

● Signals linked to function keys

● Time channels

● Objects in the alarm messages

MITSUBISHI ELECTRIC

4.2.3 How to make the communication more efficient

Group controller signals consecutively

Signals from the controller are read most rapidly if all signals in the list above are consecutive. If for example, 100 signals are defined, the quickest way to read these is to link them, for example, M0-M99 If the signals are spread (e.g. X4, Y30, M104) the updating is slower.

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 controller is to force the presentation of another block. The **New display** register can also be used if the controller is to change the block. This does not affect communication as much as a larger number of **Display** signals.

Use the clock in the terminal

An extra load is put on communication if the clock in the controller is used, since the clock register must be read up to the terminal. Downloading of the clock to the controller also creates an extra load. The interval between downloads should therefore be as long as possible.

Packaging of signals

When signals are to be transferred between the terminal and controller, they are not all transferred at the same time. They are divided into packages, each containing a number of signals, instead. The number of signals in each package depends on which driver is used.

To make communication as fast as possible the number of packages has to be minimized. Consecutive signals require a minimum number of packages, but it is perhaps not always possible to program it this way. In such cases there is a "gap" between two signals. This gap in the maximum distance between two signals which can still be kept in the same package. The size of the gap depends on which driver is used.

Signal	1	2	3	4	5	6	7	8	9	10
Used.	X	X					X	X	X	

Waste

4.3   Menu structure

The terminal is divided into three modes (function areas), configuration, programming and run mode. Programming and configuration are normally carried out with the programming tool. In each respective mode there are a number of different levels, depending on the function. Each level consists of a menu where you make a selection or enter parameters before going to the next level (menu).

The application is built up of blocks. Values from the controller are shown and changed in the blocks. Each block has a number between 0 and 989 allocated by the programmer. The blocks 990-999 are reserved for special purposes, so-called system blocks. The terminal is object-orientated, which means that a block can contain all the signals linked to an object for the control and monitoring of.
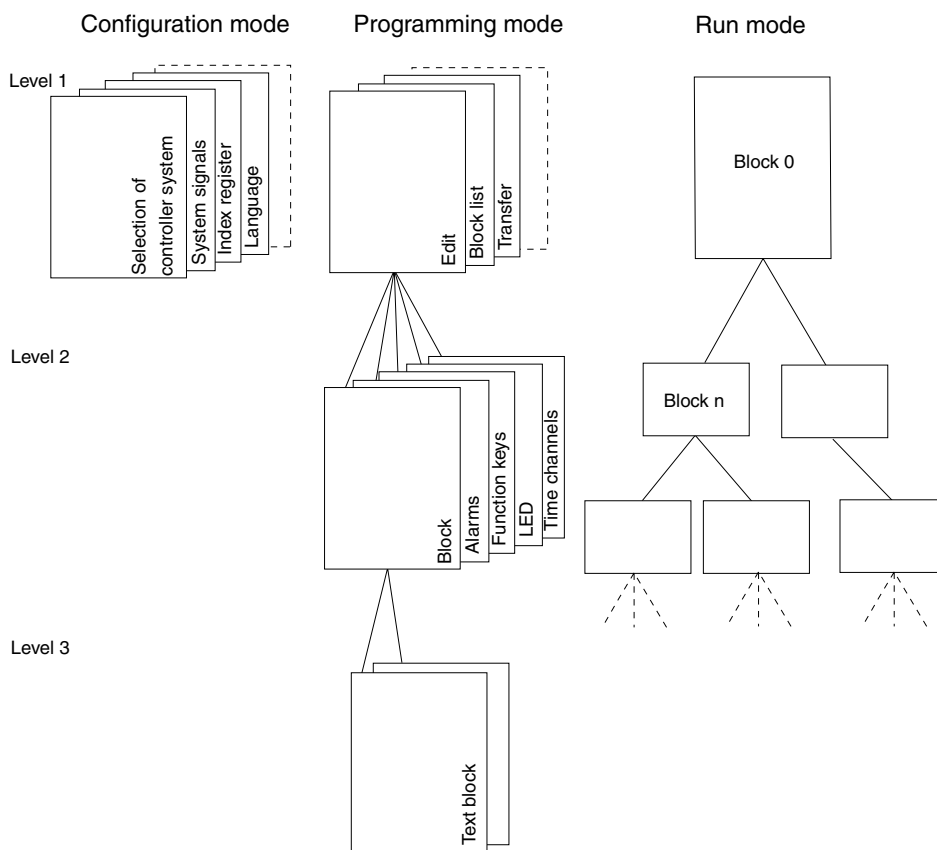
4.4   Block

A block header is defined for each block. The block numbers, type of blocks and status words etc. are specified in the block header. The functions Alarms, Time channels, System Monitor, can also be invoked as blocks. These are designated system blocks.

The maximum number of blocks in a project in E50 is 50, and in E100, E150 and E200 it is 150.

NOTE   ▮ The block type cannot be changed for a defined block.

4.5   Signal format

The following signal formats are available in the dialog for each object, on the assumption that the selected driver supports the signal format.

Format type	Area
Signed 16-bit	-32,768 – +32,767
Unsigned 16-bit	0 – +65,535
Signed 32-bit	-2147483648 – +2147483647
Unsigned 32-bit	0 – +4294967295
Float with exponent, 32-bit	±3.4E38 Numbers larger than 1,000,000 are shown with exponent.
Float without exponent, 32-bit	Parameters Positions (including decimal point and characters) and Decimals indicate the available area. E.g. 8 positions and 3 decimals result in ±999.999.
BCD Float	0 – 9999,9999.
BCD 16-bit	0 – 9999
BCD 32-bit	0 – 99999999
HEX 16-bit	0 – FFFF
HEX 32-bit	0 – FFFF FFFF

MITSUBISHI ELECTRIC

5   Programming with the programming tool

This chapter describes how the terminal is programmed by using the programming tool. For installation, refer to chapter *Installation*.

All the functions in the programming tool can be reached from the menu bar.

File Edit View Functions Setup Block Manager Transfer Window Help

5.1   Start the programming tool

Click on **Start/Program/E-Designer/E-Designer**.

The menus **File**, **Settings**, **Window** and **Help** come up in the menu bar when the programming tool is started.

5.2   Select language

Under **Settings/Menu language** you select the language the program is to be presented in, i.e. menu texts and object names, etc. In this manual we assume that you have selected **English** as the language.

5.3   Create a project

A new project is created under **File/New**. In the dialog box **Project Settings** you select the type of terminal, controller and color scheme to use in the project. All the alternatives are not available for every type of terminal. Click **OK** to create a new project.
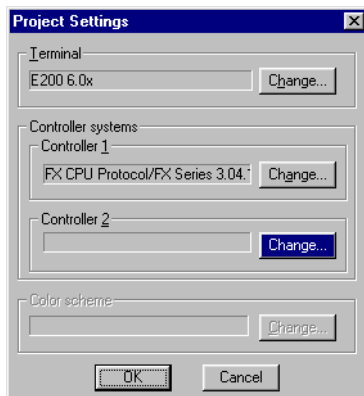
Terminal

Press **Change**.
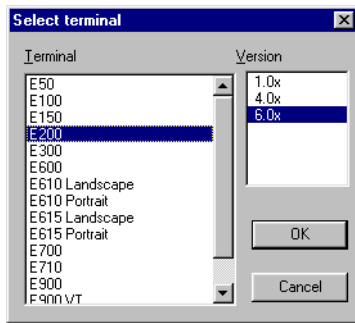
Select the terminal model and version (system program) for the actual terminal model.

Controller

The controller the terminal is to be connected to is selected here. Click the **Change** button to display the following dialog. The list shows the installed drivers. You select the manufacture, protocol and model. Click **OK** to finish. If you do not want to make any changes, click **Cancel**.
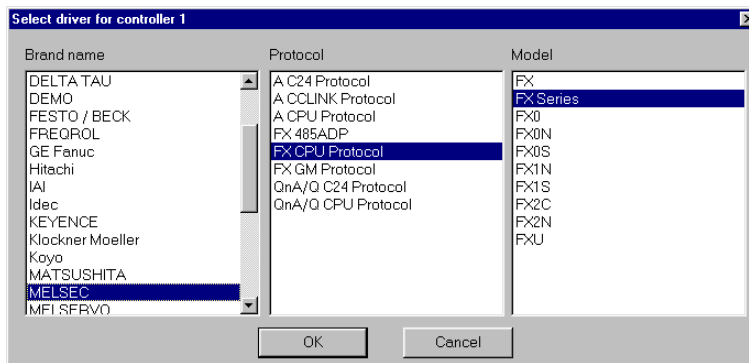
Not applicable for E50

It is possible to use two drivers in the same project, i.e. the terminal can use two drivers. The driver for the second controller is selected in the same way as for the first. For further information on two drivers in the terminal, refer to section *Communication with two controllers (Dual drivers)* in chapter *Communication*.

5.4   Update driver

5.4.1   From Internet,

To update available drivers to the latest version, or to install new drivers, you use the function **File/Update terminal drivers/from Internet**. All projects must be closed before this function is used. It must be possible to connect the computer to the Internet, but it is not necessary to have a web browser. A list of all the drivers which can be downloaded from the Internet is shown when the connection is established.
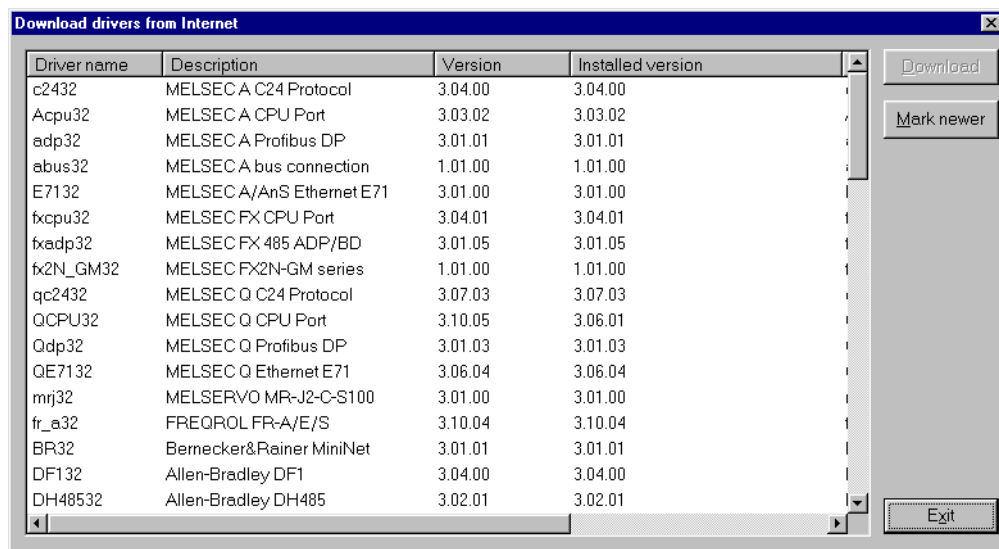
The list shows the version numbers of the available drivers and the version numbers of the installed drivers. Select the driver(s) you want to install in the programming tool. The function **Mark newer** selects all drivers available in a newer version than the one you have installed, including all the drivers which are not installed. Select **Download**. Each driver takes up approximately 500 kb and can be used as soon as downloading is complete.

5.4.2   From disk

To update available drivers to the latest version, or to install new drivers, you use the function **File/Update terminal drivers/from Disk**. All projects must be closed before this function is used. Select to open the mpd file in the catalog where the driver file is. A list of the drivers which can be installed is now shown.

The list shows the version numbers of the available drivers and the version numbers of the installed drivers. Select the driver(s) you want to install in the programming tool. The function **Mark newer** selects all drivers available in a newer version than the one you have installed, including all the drivers which are not installed. Select **Download**. Each driver takes up approximately 500 kb and can be used as soon as downloading is complete.

5.5   Change project settings

Both the selected terminal and controller can be changed for the project. Select **File/Project Settings** and click **Change** for the parameter **Terminal** and/or Controller.

5.5.1   Change terminal

If you update the system program in the terminal to a newer version you must change to the correct terminal version in the menu **Project Settings**. The new functionality in the new terminal version cannot be utilized if this is not done.

5.5.2   Change controller

If you change the controller in the project to a system with another name for the signals used, these must be changed. This is easily done by using the internal name list, see section *Name list*.

① Select **View/Name list**.

② Click on the button **Undefined** to insert all I/Os used in the project in the name list.

③ Click on the button **Export** to export the name list to a text file. Enter a name and click **Save**. Define how the text file is to be separated.

④ Open the text file in a text editor, e.g. Wordpad.

⑤ Change all I/O to signals used in the new controller, and sabe the file as a text file.

⑥ Click on the button **Import** in the dialog Name list, and click **No** to the question if invalid I/Os should be removed.

⑦ Click on the button **Rebind** to update all new I/Os in the project with the new names.

⑧ Select **File/Project Settings** and click **Change**.

⑨ Select the new controller to be used and click **OK** twice.

5.6   Creating a block with the Block Manager

When a project is created, the Block Manager is shown. All blocks included in the application are presented here. The main block, block number 0, is created automatically when a new project is created.
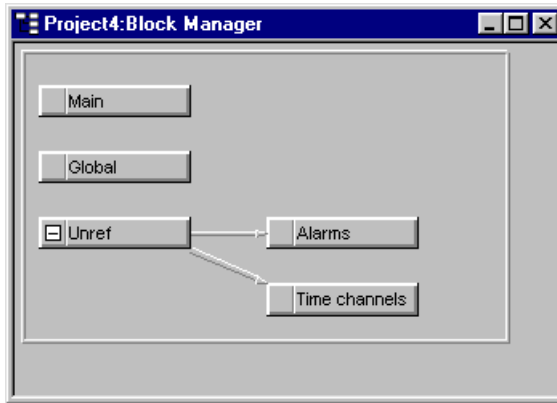
The Block Manager includes a toolbox containing the following functions.

Selects blocks and creates new blocks.

Adds a new block.

Defines the block header for the selected block.

Makes a jump to the new block to be created with the cursor.

Makes a jump with the function key to the new block to be created with the cursor.

Adds a block jump for the pointer key.

Deletes a selected block.

Opens the selected block for editing.

Setting for the Block Manager.

Zoom in

Zoom out

5.6.1   Define block

When you have decided to add on a block, the following dialog is shown. This is a simplification of the complete block header. When you click on OK, i.e. create a block, the block is shown/opened.
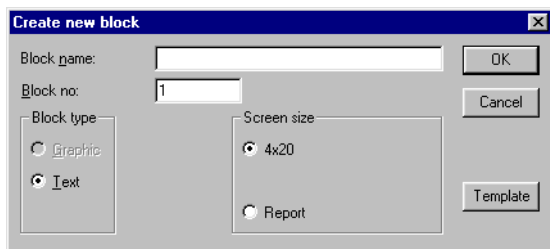
Block name:

A name for the block can be entered here. The block name is shown in the Block Manager and in the block list.

Block no:

The number of the block. If a block already exists with the number given, the defined values will be filled in automatically. Block 0 is created automatically when starting and must always be used in a project.

Block type

Select a text block.

Block width

Select the character size for the text block. The character size cannot be changed for a defined object.

Template

Here you can copy a block template to the block, or save the block as a block template.

◆ MITSUBISHI ELECTRIC

5.6.2   Define the complete block header

The default parameters applicable for each separate block are defined in the block header. Select a block by clicking on it, and select **Block Manager/Header**. In the **Block Header** the following parameters can be defined:

Block no:

The number of the block. If a block already exists with the number given, the defined values will be filled in automatically. Block number 0 is created automatically when starting and must always be used in a project.

Block name:

A name for the block can be entered here. The block name is shown in the block list.

Display signal:

Not applicable for E50. A digital signal which, when activated, results in the block being shown on the terminal's display. To change blocks as quickly as possible, display signals in consecutive order should be used. This field is not filled in if another method of changing blocks is used.

Print signal:

Not applicable for E50. A digital signal which, when activated, results in the block being printed out on a connected printer. **Display signal** and **Print signal** can be the same. Print signals in consecutive order should be used to obtain the fastest possible printouts.

Completion signal:

Not applicable for E50. A digital signal activated by the terminal when the printout is ready. The signal is normally set to one. If the box **Reset** is marked, the signal will be reset when the printout is ready.

Recipe directory

Not applicable for E50. Select a recipe directory in which all recipes created in the block will be saved. See chapter *Recipe handling*.

Security Level:

Not applicable for E50. A security level (0-8) for the block is given here. If a security level of > 0 is given, the operator must log-in with a password corresponding to the given security level or higher.

Block width:

Only shown if the block is a text block. Select the character size for the text block. It is not possible to change the character size for a defined object in a text block.

Fkeys:

Local function keys for the block are defined here. See also chapter *Function keys.*

Keyfield:

Not applicable for E50. The lead texts for the functions keys are defined here. Enter an optional text of a maximum of six characters in E200 and a maximum of five characters in E100 and E150. The bottom line in the block is used.

Template:

Here you can copy a block template to the block, or save the block as a block template.

Status:

Indicates the status of the display in run mode. The status word does not affect the system blocks. The parameters in the status word have the following functions.

Cursor off:
Not applicable for E50. Indicates if the cursor is to be visible in the block in run mode.

Put cursor at first man. object:
Not applicable for E50. Indicates if the cursor is to go to the first maneuvrable object in the block instead of at the top left.

Disable main menu key:
Only applicable for E200. Indicates if the [MAIN] key is to be disabled in run mode when the block is visible on the display.

Disable alarm list key:
Only applicable for E200. Indicates if the [LIST] key is to be disabled in run mode when the block is visible on the display.

More indicator:
Indicates if the + character is to be shown at the bottom and top right on the display when there is more in the block below or above what can be seen on the display.

Auto data entry:
Not applicable for E50. Indicates if the cursor is to move automatically to the next maneuvrable object after an input. The cursor can only be placed on maneuvrable objects in this position.

Disable PREV function:
Only applicable for E200. Indicates if the [PREV] key and the function "Return to previous block" for the function keys is to be disabled in run mode when the block can be seen on the display.

Disable ENTER function:
Only applicable for E200. Only valid for digital objects. Indicates if the [ENTER] key is to be disabled in run mode when the block can be seen on the display.

5.7 Show terminal around the working area

For the terminals there are the functions **View/Options/Show Terminal**. When this function is active an image of the actual terminal is drawn around the working area on the active block. The terminal image has pressure-sensitive surfaces on the function keys, LEDs and text fields.

5.7.1 Define function keys

When you double-click on a function key you can select to define a local or global function key. The Editor for the selected function is then shown. For definition of function keys, see chapter *Function keys.*

5.7.2 Define LEDs

The Editor for definition of LEDs is shown when you double-click on an LED. For definition of LEDs, see chapter *LEDs*.

5.7.3 Create text strips

If you double click on a field in the text strip a dialog will be shown where you can enter text, select adjust text, or select the font. This function enables complete text strips to be made and printed out on plastic film, which can then be used on the terminal.

5.8 I/O Browser

If you create a local name list in your project you can select I/O signals from this name list with the button **I/O** when you define your objects. The button **I/O** is included in all dialogs which allow you to give an address. The I/O Browser has incremental searching. This means that searching begins immediately when you start entering in the characters in the Name/Address field. The I/O list is sorted by signals or names.
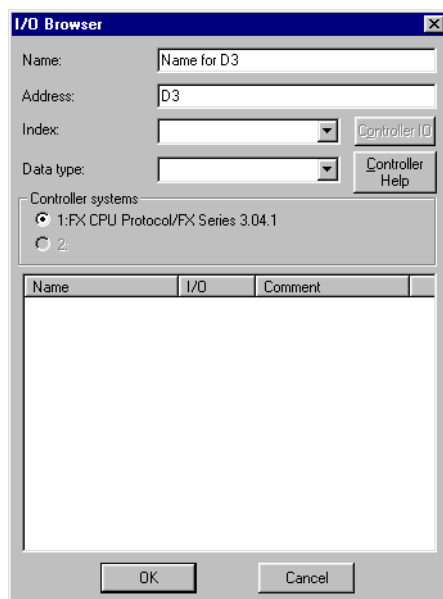
5.9   Programming blocks

Double click on the required block in the **Block Manager**. The working area for the block and the toolbox are now shown. The objects which can be created in the block are in the toolbox.

Select an object by clicking on the object in the toolbox. Click to activate the dialog box for the selected object. Fill in the parameters in the dialog box, and click **OK**. The object is now shown in the working area.

General parameters for the objects are described in the chapter *General*. The chapter *Text-based presentation and maneuvring* explain text objects.

⯅ **MITSUBISHI ELECTRIC**
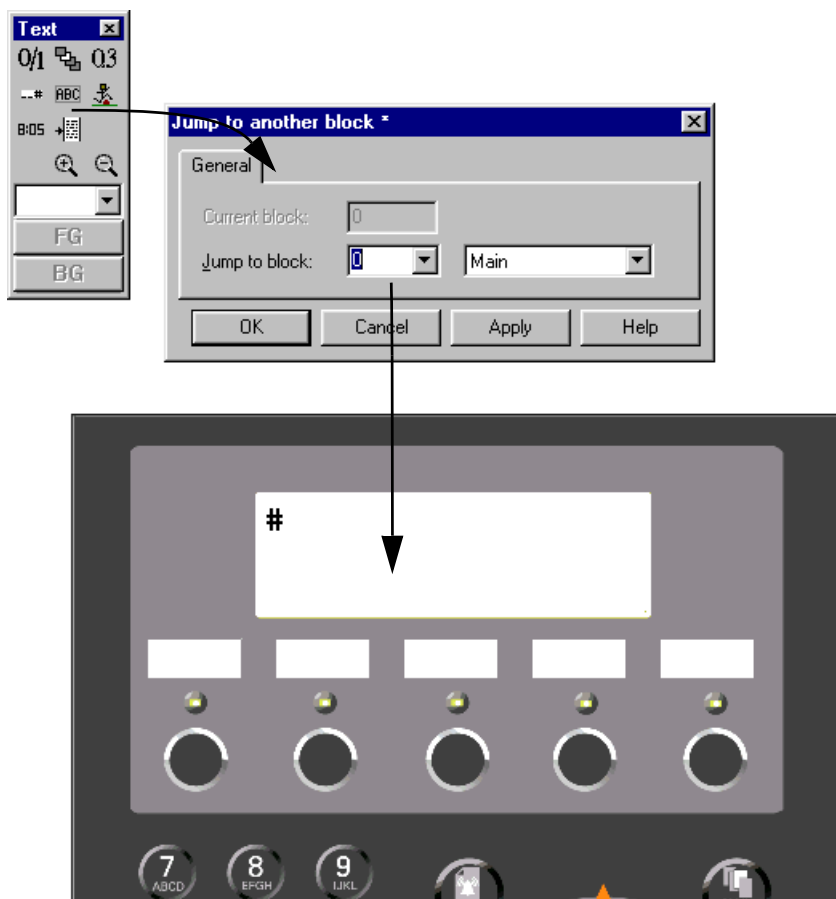
5.10   Text block Manager

In Text Block Manager, dialogs and reports are built up. A text block can consist of static texts and dynamic objects. Static texts are texts which cannot be changed during the execution of the program. Dynamic objects are objects which can be linked to signals in the controller. There are seven types of dynamic objects – digital, analog, jump, date/time, column, multiple choice and text objects.

5.10.1   Open Text Block Manager

The manager opens when you double click on a defined text block in the Block Manager or in the block list. In the Block List you select a defined block, or create a new text block.

5.10.2   Mouse and keys

Click on the beginning of the text to be marked, and drag the mouse pointer over the text. Press the shift and arrow key at the same time to select the text if using the keyboard.

The cut function erases selected text.

The character End of line is inserted by pressing [Ctrl] [Enter].

Double click or press [F4] on an object to show the object's parameters.

5.10.3   Toolbox

In Text Block Manager there is a toolbox with the following functions. All functions are also available in the menus. See chapter *Text-based presentation and maneuvring* for information on how the different objects are defined and used in text blocks.

Zoom in

Zoom out
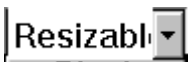
ASCII options list. Selection of characters
not available on the keyboard.

5.10.4   Define text block

Static text

Text Block Manager is a text editor where static text is entered. The Windows functions copy/paste can be used to copy/paste text in or between blocks, or between other Windows programs such as Microsoft Word. This enables simple documentation of the application.

Dynamic objects

Dynamic objects can be defined anywhere in the text. Select the type of object in the toolbox or in the menu **Object**. A dialog is then shown where the object is defined.

The dynamic object is indicated with the character # followed by one or more '-' depending on the number of positions the object occupies. For more detailed information of the definition of dynamic objects, refer to chapter *Text-based presentation and maneuvring.*

5.11   Change I/O

The function **Change I/O** allows you to change an I/O to another, or move a complete I/O area. I/O can be changed in a complete project, or only in the objects marked when the function is selected.

The function can be used on blocks in the block list, objects in text blocks, lines in the alarm list, lines in the Function Key Editor, lines in the LED Manager and lines in the cross-reference list.

Select the function **Edit/I/O Change**.
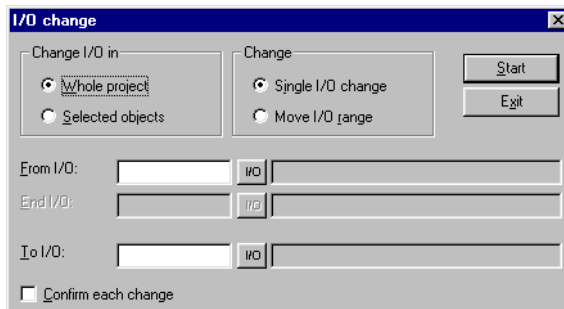
Change I/O in

Select whether you want to change I/Os in the whole project or for selected objects.

Change

Select whether one I/O is to be changed, or whether you want to move a complete I/O area.

From I/O, End I/O, To I/O

You indicate here the I/O to be changed, and to what or which I/O area it is to be moved.

Confirm each change

Select whether you want to confirm changing the I/O for each object.

5.12 Change BDTP station

This function allows you to change the index numbering in a BDTP client project included in a BDTP network. For example, from station 1 to station 3. Select the function **Edit/BDTP Station Change**.
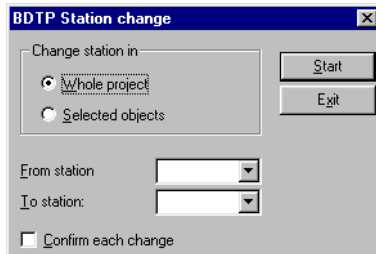
Change station in

Select whether you want to change the index numbering in the whole project or for marked objects.

From station, To station

The index number to be changed is indicated here, and to which BDTP station index number it is to be changed to.

Confirm each change

Select if you want to confirm changing the BDTP station for each object.

5.13 I/O Crossreference

The function crossreference is used to document I/Os in a well-arranged way. Select the function **View/I/O crossreference**.
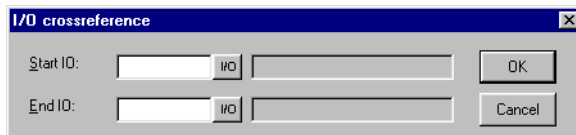
In the dialog shown you can indicate **Start I/O** and **End I/O.** If the **Start I/O** field is left empty all I/Os to the value in the **End I/O** field will be included. If the **End I/O** field is left empty all I/Os from the value in the **Start I/O** field will be included. If both fields are left empty all I/Os will be included in the list.

5.13.1 Presentation

The results of the function are presented in a list with two levels. The first level shows which I/Os are used and how many objects refer to each respective I/O.

The second level is reached by double clicking on the + character to the left of the I/O. The objects containing the I/O you have clicked on will now be shown. The + character becomes a minus character.
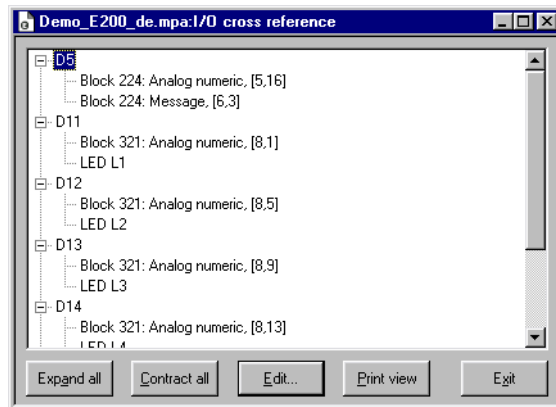
A line in the list can be marked and copied to the clipboard, from where it can be pasted in, for example, a Microsoft Word document.

MITSUBISHI ELECTRIC

5.14   Other managers/editors

The programming tool also has managers/editors for function keys, LEDs, alarms, alarm groups, passwords, time channels, message libraries, macros, and data transfers. These managers/editors are activated from the menu **Functions** and are handled in a similar way. The parameters in respective managers are described in the respective chapters.

Defined function keys, LEDs, alarms, alarm groups, time channels, message libraries, macros and data transfers are shown in a list in respective editors. Add on new definitions with **Append** or **Insert**. Change a definition by selecting it, making the changes, and clicking **Update**. To simplify the handling of more than one definition, click **Update** or **Append** the first time, and then [RETURN]. The functions **Append** and **Update** are active until the other is activated. With **Delete** or **Clear** a selected definition can be removed. Exit from the manager with **Exit**. The following example describes the Alarms Manager.

The alarms are numbered automatically. If **Append** is used the alarm definition will be placed last in the alarm list, and if **Insert** is used the new definition will be placed on top of the marked line in the list. Subsequent alarm definitions will be renumbered. Changes made are confirmed by clicking **Update**.

5.15   The File menu

In the menu **File** there are functions to create, open, save and close projects. Printer settings, preview printout, create document header, and activate printout are also selected from here. There are also functions to test the project and change project settings. With the function **MELSEC I/O Naming** you select whether an internal name list is to be used, or if a name list from GX Developer or GX IEC Developer is to be imported. The function can only be used if a driver for MELSEC PLC system from Mitsubishi Electric is used.

```
File  Edit  View  Functions  Setup  Block Mana
  New                              Ctrl+N
  Open...                          Ctrl+O
  Close
  Save                             Ctrl+S
  Save As...

  MELSEC I/O Naming...

  Update terminal drivers              ▶

  Project Test
  Project Settings...
  Document Header...

  Print...                         Ctrl+P
  Print Preview
  Print Setup...

  Exit
```

5.16   The Edit menu

In the menu **Edit** there are functions for cutting, copying, pasting, undoing and selecting all. There are also functions to Change I/Os, BDTP Station and Select default controller.

```
Edit  View  Functions  Setup  Block
  Cut                    Ctrl+X
  Copy                   Ctrl+C
  Paste                  Ctrl+V
  Undo                   Ctrl+Z

  Select all             Ctrl+A

  I/O Change...
  BDTP Station change...
  Default controller            ▶
```
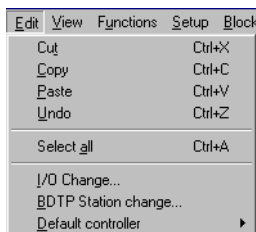
5.17   The View menu

In the menu **View** there are function managers for block handling, crossreference and name list. There are also functions to set different viewing alternatives in the program. Some functions are standard for Windows and some are unique for the programming tool. A description is provided here of those unique for the programming tool.

```
View  Functions  Setup  Block Ma
  Block List          Ctrl+B
  Block Manager       Ctrl+M
  Symbol Manager...   Ctrl+Y
  I/O crossreference...
  Name list...

  Toolbars              ▶
  Options               ▶
  Zoom                  ▶
```

🔶 **MITSUBISHI ELECTRIC**
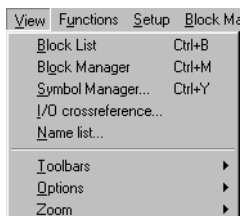
5.17.1   Block List

The block list shows which blocks are included in the application. New blocks are created from the block list with the button **New** and defined blocks are opened with **Open**. The block header is shown when you select New. The default parameters for the block are defined in the block header. You can open the dialog for the block header for blocks marked in the list by pressing the button **Header**. Remove marked blocks with the button **Delete**.
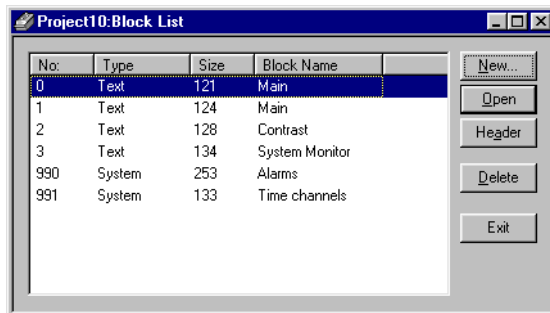
5.17.2   Block Manager

The Block Manager shows which blocks are included in the application. You can create new blocks here, define the block header and define jumps with the functions in the toolbox.

5.17.3   I/O crossreference

Under **I/O crossreference** you can document I/Os in a well-arranged way. See section *I/O Crossreference*.

5.17.4 Name list

Under **Name list** you define a local name list for the signals used. Signals in the project which are not named can be inserted in the name list with the function **Undefined**. New signals can be added and existing names can be edited and updated. The function **Update** updates the project with the changes made in the name list.

A name list can be exported to a text file, and a text file can be imported to a name list. The contents in the file can be separated either with a semicolon, comma or space. An internal name list can be sorted. The text file should not contain national characters

NOTE ▌ In the function MELSEC I/O Naming in the **File** menu you must select **Internal name list only** to use the name list. This function only functions if drivers for MELSEC PLC systems are used.
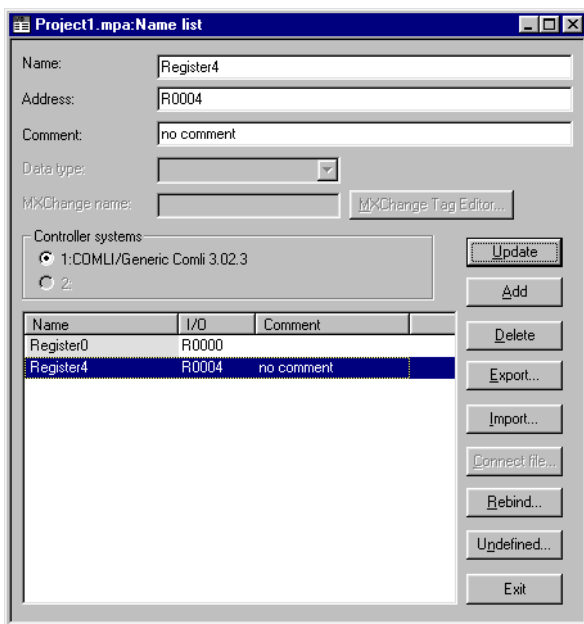
If you have linked a driver-specific name list to your project you can select I/O signals from this name list with the button **Connect file**.

5.17.5 Toolbars

Under toolbars you can show/hide different toolbars in the program.

Toolbar

Select to show the toolbar.

Controller toolbar

Select to show toolbar for currently selected controller.

Language toolbar

Select to show toolbar for currently selected application language.

Status Bar

Select to show the status bar.

Toolbox Block Manager

Select to show toolbox for Block Manager.

Toolbox

Select to show the toolbox.

Align Toolbox

Show toolbox for the align functions.

5.17.6 Options

Show language index

Applicable only when multiple-language support is used. Displays index number for text in the application.

Tool tips

Show lead texts for the function the cursor points to.

Use block list

You can select whether the program is to activate the block list or the Block Manager when you a new project is created.
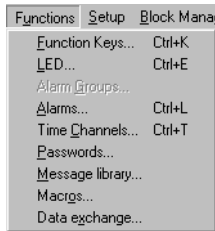
Use terminal font

You can select whether the program text you enter in the dialog windows is to be shown with terminal font.

Use Unicode font

Choose a Unicode font in the font dialog box. The font is used in the programming tool when multiple-language support is used.

5.18 The Functions menu

In the menu **Functions** there are function managers for function keys, LEDs, alarm groups, alarms, time channels, passwords, message libraries, macros and data exchange.

```
Functions  Setup  Block Manag
  Function Keys...    Ctrl+K
  LED...              Ctrl+E
  Alarm Groups...
  Alarms...           Ctrl+L
  Time Channels...    Ctrl+T
  Passwords...
  Message library...
  Macros...
  Data exchange...
```

5.18.1 Function Keys

Global and local function keys are defined under **Function keys**. See chapter *Function keys*.

5.18.2 LED

Not applicable for E50 – 150. Under **LED** you define the functions for the LEDs.
See chapter *LEDs.*

5.18.3 Alarm Groups

Not applicable for E50 – 150. Under **Alarm Groups** you can group alarms, e.g. according to degree of seriousness, so that they can be identified and rectified more quickly and easily. See chapter *Alarm handling*.

5.18.4 Alarms

Not applicable for E50 – 150. Alarm messages and which signals are to activate the alarms are defined under **Alarms**. See chapter *Alarm handling.*.

5.18.5 Time Channels

Not applicable for E50. Under **Time Channels** you define time channels to be used to control events in the process at special times. See chapter *Time control.*

5.18.6 Passwords

Not applicable for E50. Under **Passwords** you define passwords for the different security levels in the application. See chapter *Password.*

5.18.7 Message library

Not applicable for E50. Under **Message library** you create message tables, where values between 0 and 65,535 are linked to texts. See chapter *Message library*.
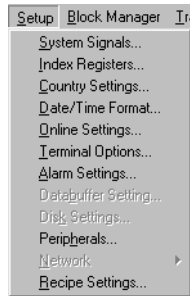
5.18.8 Macros

Not applicable for E50. Under **Macros** you create events which influence all function keys and pointer keys. See chapter *Macro*.

5.18.9 Data exchange

Not applicable for E50. Under **Data exchange** you define which conditions data exchange should utilize between selected controllers. See chapter *Communication*.

5.19   The Setup menu

In the menu **Setup** there are functions to configure the terminal.

Setup Block Manager Tr
 System Signals...
 Index Registers...
 Country Settings...
 Date/Time Format...
 Online Settings...
 Terminal Options...
 Alarm Settings...
 Databuffer Setting...
 Disk Settings...
 Peripherals...
 Network   ▶
 Recipe Settings...

5.19.1   System Signals

Setting of handshake signals between the terminal and the controller.

Current display reg

Data register in the controller, which in run mode contains the number of the block shown on the display. The data register is updated automatically by the terminal when changing blocks. This register does not affect the selection of blocks.

New display reg

A data register in the controller, which controls which block is to be shown on the display.

Backlight signal

Digital signal which controls whether the display is to be switched on or off.

Printer status reg:

Not applicable for E50. You can read the status of a printer connected to the terminal via the expansion card for parallel printer through a register. The register can assume the following values.

Registerwert	Description
0	OK. The printer is functioning as it should.
1	General error. Check port and printer settings.
2	No paper. Put paper in the printer.
3	No memory. The printer's memory is full.
4	Not connected. The printer is not correctly connected. Check port and printer settings, including the cable.

If the printer status register has the value 1-4, i.e. does not function normally, the terminal will ignore all printouts until the register assumes the value zero.

Library index reg:

Not applicable for E50. Used for indexing of the message libraries. The number of the library the texts are to be collected from is indicated in the message object. If an index register is defined, the content in the index register is added to the number given in the object. This makes it possible to control from which library the texts are to be taken with a register.

Commands

On the commands line it is possible to give one or more of the following commands. The commands are typed with all capital letters. The commands are separated by a space.

Command	Description	Supported by
BR	BatchRead. Forces terminal into Transparent mode communication when MELSEC controller used.	E50-E200
BSD	Baud Switching Disable for Transparent mode communication with FX2N and Q. **BSD** must be set in the terminal that is not connected to controller 2.	E50-E200
Rx	Maximum number of re-sends. **x**=total tries. Applies to communication with the controller. Example: R5@2 applies to controller 2.	E50-E200
Tx	Global timeout in x total ms. Applies to communication with the controller. Example: T10000@1 applies to Timeout for controller 1.	E50-E200
AKx	Activates Joystick function. See the section *Joystick-Funktion.*	E100-E200
DD	Disable Delete of alarms on alarm list. When specified, inactive and acknowledged alarms cannot be deleted from the alarm list.	E100-E200
LOBx	Replaces the digital signal **x** when the real-time clock's battery requires replacement. Example: **LOBM0** will reset M0 when the battery requires replacement.	E100-E200
MDx	When dual drivers are used: If communication to a controller fails, the terminal will continue to communicate with the other system. The terminal will try to connect to disconnected system every tenth second. The command **MDx** is used to change the interval; **x** is the time in ms.	E100-E200
NTx	Timeout in **x** number of ms for messages in No protocol mode.	E100-E200
RPD	RUN/PROG DISABLE. Disables capability to switch between RUN/PROG with the [BACKSPACE] and [MAIN] keys. If **RPD** is used, the terminal can only switch modes via the programming tool.	E100-E200
SW	Converts text with Swedish ASCII (7-bit) characters to 8-bit IBM PC extended ASCII when printing.	E100-E200
NTGL	NUM key assigned toggle function. NUM key activates/deactivates input of numeric characters on E100.	E100

5.19.2 Index Registers

Indexadressierung dynamischer Objekte. Nähere Informationen entnehmen Sie dem Kapitel *Indexadressierung*.

5.19.3 Country Settings

Character set:

The selection of the character set will influence the character table used in the terminal and which national characters will be available.

In text-based terminals, special character tables are used and the same character table will be used irrespective of which character set is selected.

Language:

Select menu language in the terminal; *British English*, *German*, *Swedish* or *American English*. The default setting for menu texts in the terminal is British English.

5.19.4 Multi language

New language

Start the wizard to create multiple languages for an application.

Edit

Here you can edit translated texts in the application languages.

Setup

The structure tree is displayed for the languages in the application. For information on the settings that can be made, see the chapter *Language management*.

Export

This function exports the user texts to a text file in ANSI, OEM or Unicode format. Choose to export either user texts or system texts. After making your choice, the **Export Multi language texts** dialog box is displayed. Here you can specify where the file will be saved and its format. Under **Encoding**, you can choose **ANSI/OEM**, which entails that languages created in the ANSI/OEM format will be exported, or **Unicode**, which entails that all languages will be exported to a file in Unicode format.

Import

This function imports a language that can subsequently used in the terminal. Choose to import either user texts or system texts. After making your choice, the **Import Multi language texts** dialog box is displayed. Indicate the name of the text file that you want to import. If the existing language in the project is in ANSI/OEM format and a language in Unicode format is imported, the imported language is converted to ANSI/OEM. This entails that all characters not included in ANSI/OEM are displayed as question marks.

Show index

This function is for displaying the language index instead of texts in the object. You may enter text even when though the language index is shown. The new text will then receive a new language index.

Cross reference

Shows a cross-reference list of the indexes for the various blocks in the application.

Reuse index

If this function is active when an object is copied, a new object is created with the same index.

Choose Unicode font

Choose the Unicode font that will be used in the programming tool.

5.19.5   Date/Time Format

Not applicable for E50.

Setting of the date and time format.

Date format:

The format can be YY-MM-DD, YYMMDD, DD.MM.YY, DD/MM/YY or
MM/DD/YY where Y=year, M=month and D=day.

Time format:

The format can be HH:MM:SS or HH:MM where H=hour, M=minute and S=second.

Clock used:

Terminal means that the terminal's built-in clock is used, and controller 1 or 2 means that the
clock in controller 1 or 2 is used.

Clock→Controller system 1/2:

Select if the clock in the terminal is to be transferred to a data register in the controller 1 or 2.

NOTE   ▌ In those cases where the controller has an activated real-time clock and the terminal clock is
             to be transferred to the same data register, the controller clock has precedence.

Update interval:

Indicate how often the terminal is to transfer the clock data to the controller. The value is given
in seconds, and 60 seconds is recommended. A shorter updating interval will make communi-
cation between the terminal and the controller slower.

Controller register:

Enter the start address for storage of the date/time in the controller.

For information on how the date and time is saved, refer to the manual for the controller used.
The terminal saves information in the same order as the default setting for the controller.

Daylight saving:

The start and end of summer or daylight saving time are defined here. State the Day of week,
Week In Month, Month, Hour and Adjust. You can choose between European and US standard.

To disable the function for summer time leave both fields for the month blank.

5.19.6   Online Settings:

Not applicable for E50.

Permits changing of selected function in RUN mode.

5.19.7 Terminal Options

Not applicable for E50.

Screen save time:

The time in minutes after the last event on the display when it will switch off. The default setting is 0 which means that the display never switches off. This function optimizes the service-life of the display.

Key delay time (ms):

The time interval in ms between pressing the same key before the cursor automatically moves to the next position. Used for input of ASCII characters (A-Z etc.). See section Alphanumeric keys in chapter Terminal functions.

Key beep:

Indicates whether the terminal will beep when a key is pressed.

Key repeat:

Indicates if a function will repeat as long as the key is pressed. Function keys and input of alpha-numeric characters (A-Z etc.) are not repeated.
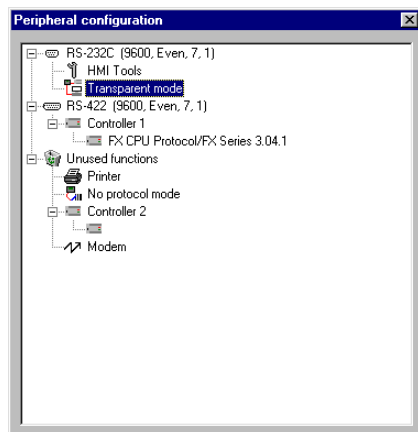
5.19.8 Alarm settings

Not applicable for E50.

General settings for alarm list. For further information, see chapter *Alarm handling.*

5.19.9 Peripherals

All communication configuring is made under **Setup/Peripherals**. The units defined for the system are shown when you select **Peripherals**. You can use the functions drag and drop to move the units.

Edit

You can select a unit or port on the display and click **Edit** or click with the right mouse button. The dialog for the marked unit/port will be shown.

Ports

When you select **Ports** a dialog is shown where you can see the current configurations. The configuration can also be changed here.

NOTE   ▌ The highest transfer speed when E50, E100 or E150 are used is 38,400 baud.

RS232C

When you select the **RS-232C** port and and right-click, the following dialog box is displayed. Select the transfer speed, parity, number of data bits and stop bits for the port. In order to use RS-232C on E50 an adapter must be connected to the RS-422 port.
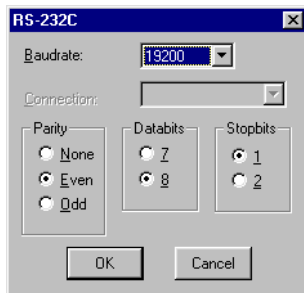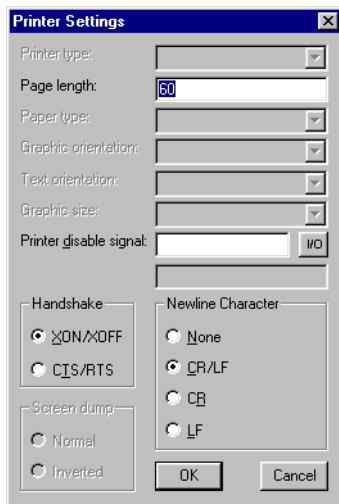
RS422

When you select the **RS-422** port and right-click, the following dialog box is displayed. Select the transfer speed, parity, number of data bits and stop bits for the port.
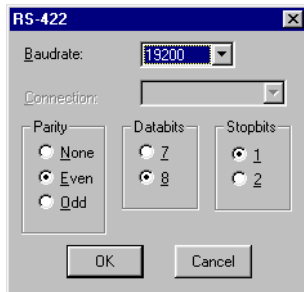
Printer

Not applicable for E50.

When you select **Printer** and click **Edit** the dialog for the printer settings is shown.

Printer type:

Select printer, None, HP PCL5 or Standard text.

MITSUBISHI ELECTRIC

Page length:

The number of lines to be printed out before form feed. If the page length is set 0 there will be no form feed. The default setting is 60.

Paper type:

Select type of paper.

Text orientation:

Select whether the report printout is to be vertical or horizontal.

Printer disable signal:

Digital signal to stop the printout.

Handshake:

Select the required type of handshaking between the printer and terminal; XON/XOFF or CTS/RTS. Refer to the printer manual for information on suitable handshaking.

Newline Character:

Select the required line end/newline character; CR/LF, CR, LF or none.
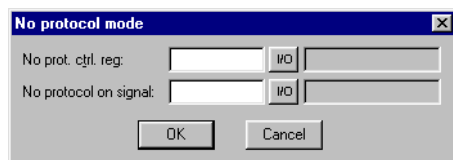
Screen dump:

Printout of the screen. You can select Normal or Inverted printout.

NOTE   ▌ The printer settings are used for parameters such as character set, character size and margins.

No protocol mode

When you select **No protocol mode** and click **Edit**,or click with the right mouse button, the following dialog is shown.

No prot. ctrl reg:

The first control register used in No protocol mode. No protocol mode is described in chapter *Communication*.

No protocol on signal:

A digital signal for switching between No protocol mode and Transparent mode. Used for switching between these modes in run mode, for example to call a computer and send a message.
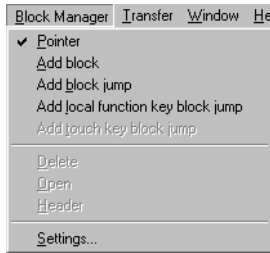
5.19.10   Recipe settings

Not applicable for E50.

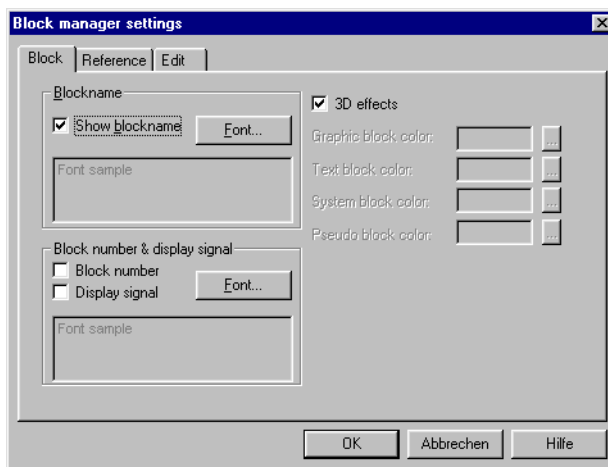Settings for recipe handling are made here. See chapter *Recipe handling.*

5.20   The Block Manager menu

In the menu **Block Manager** there are functions for programming blocks.

5.20.1   Settings for Block Manager

Under **Block Manager/Settings** there are settings for presentation in the Block Manager.

Block:

Select how data for the block is to be presented, including the layout for the block in the Block Manager.
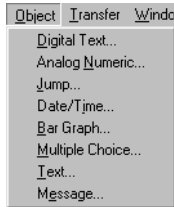
Reference:

Select overview layout for the Block Manager.

Edit:

Select special functions for the presentation in the Block Manager.

5.21   The Object menu

The menu **Object** includes all objects available in the program. The number of objects varies depending on the type of terminal. See description of each object in chapters *Graphic presentation and maneuvring and Text-based presentation and maneuvring*.

Object Transfer Windo
  Digital Text...
  Analog Numeric...
  Jump...
  Date/Time...
  Bar Graph...
  Multiple Choice...
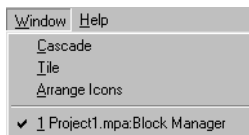 Text...
  Message...

5.22   The Transfer menu

In the menu **Transfer** there are functions for transferring projects, selected blocks, and communication settings for transfer between PCs and the terminal. See chapter Transferring projects.

Transfer Window Help
  Project...
  Comm Settings...

NOTE   ▮ The communication settings must be same in the programming tool as in the terminal.

5.23   The Window menu

The menu Window includes all the functions standard to Windows.

Window Help
  Cascade
  Tile
  Arrange Icons
  ✔ 1 Project1.mpa:Block Manager

5.24   The Help menu

The **Help** menu contains help texts and information on the program version.

Help
  Tip of the Day...

  Index
  Using Help

  Controller Help

  About E-Designer

MITSUBISHI ELECTRIC

6   Text-based presentation and maneuvring

Text-based presentation and maneuvring are appropriately used for the generation of different types of report printouts. These can be daily reports or status reports, etc. Reports are built up text blocks, which can contain both static and dynamic text. For examples of what a report can look like, refer to chapter *Printouts*. This chapter presents text objects in tabular form. Every object is then described separately.

NOTE   ▌ Text-based printouts are not supported when Unicode is used.

6.1   General parameters

6.1.1   Scaling

Not applicable for E50

The parameters **Offset** and **Gain** are used to change the scale of the controller value to a value shown in accordance with the following equation.

Panel value = Offset+Gain*Controller value

When the value for an object is changed from the terminal in run mode, the scale is changed to the value shown in accordance with the following equation.

Controller value = (Panel value-Offset)/Gain

Scaling does not affect the defined maximum and minimum values, or the number of decimal places.

NOTE   ▌ The functions Increment and Decrement affect the Controller value for the maneuvrable object, and not the Panel value.

6.1.2   Offset Gain Calculation

The function **Offset Gain Calculation** is a tool to calculate the Offset and Gain parameters. Specify the value of Offset and Gain under General for the object and click on **Calculation**. The following dialog is now shown.


Enter the interval for the controller value and panel value. The function calculates the correct values for the parameters **Offset** and **Gain**.

6.1.3   Access
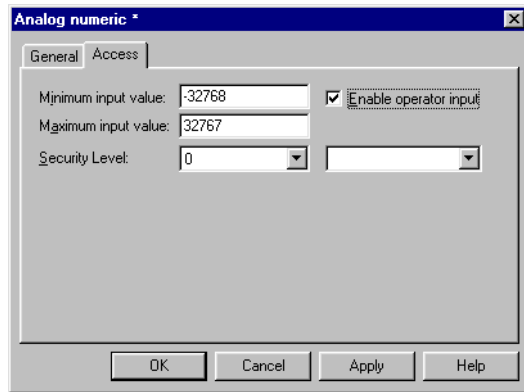
Under **Access** you specify whether the object is to be maneuvrable, and the **Minimum** and **Maximum input value** for the object (and maneuvring). You can also select the security level for the object. The security levels are defined under **Functions/Passwords**. See chapter *Password*.

For the object **Message** the dialog appears as follows.
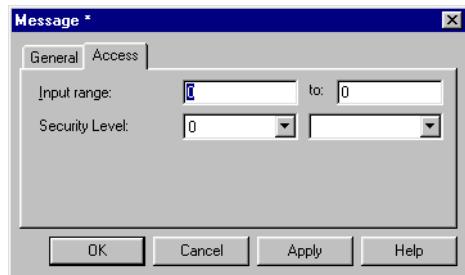
Specify the **Input range** the first and last text in the area. An area of a maximum of 64 texts can be maneuvred in run mode.

6.2 Text object

6.2.1 Dynamic text object

Symbol	Object	Description
0/1	Digital text	Switches between two texts depending on the status of a digital signal.
	Multiple choice	Links to a data register, which can assume eight different states. A text of up to 30 characters can be linked to each state. Not E50.
0.3	Analog numeric	Presents values in numeric form.
--#	Bargraph	Presents values in the form of bar diagrams. Not E50.
ABC	Text	Handles ASCII strings. Not E50.
	Jump	Jump to another block.
8:05	Date/Time	Setting of date and time. Not E50.
	Message	An object which shows texts from a Message Library. Not E50.

0/1 Digital text

Text object used to switch between two entered texts depending on the status of a digital signal.
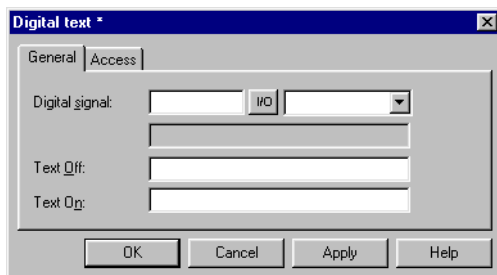
Parameter	Description
Digital signal	Address of the signal. See section *General parameters*.
Text Off	The text shown when the signal is zero (off).
Text On	The text shown when the signal is one (on).

Access

The functions under **Access** are described in section *General parameters*.

🔲 Multiple choice

The object is linked to an analog signal, which can assume 8 different states. A text of up to 30 characters can be linked to each state.
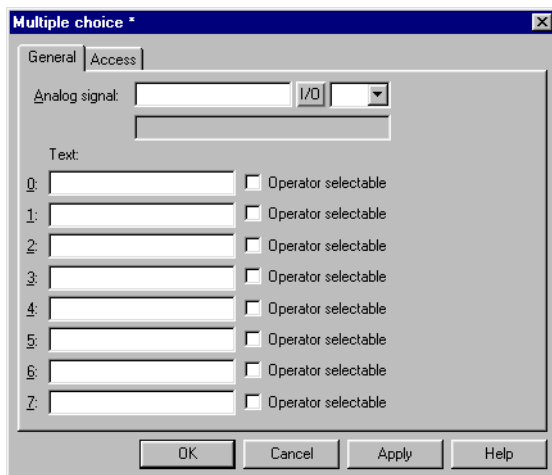
Parameter	Description
Analog signal	The register which determines which text is to be shown.
Text 0-7	The texts to be shown for respective states of the object.
Man. 0-7	Indicates if it is possible to maneuvre to this status in run mode. See section *Maneuvre text block*.

Access

The functions under **Access** are described in section *General parameters*.

0.3 Analog numeric

An object which presents values in numeric form, as whole or decimal numbers.

🔺 **MITSUBISHI ELECTRIC**
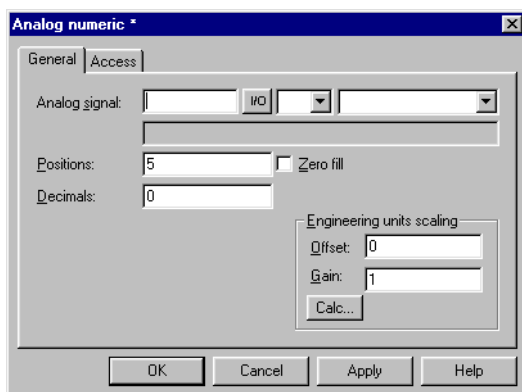
General

Parameter	Description
Analog signal	Address of the signal.
Positions	The number of positions the object uses. Characters such as minus signs or decimal points must be included.
Zero fill	Select whether zeros are to be written in empty positions.
Decimals	The number of decimals the object is to be presented with.
Engineering units scaling	Indicates the scaling of the value, see section *General parameters*.

**Access**

The functions under **Access** are described in section *General parameters*.

--#   Bargraph

Presents values in the form of bargraphs. The bar will be shown horizontally so that the maximum number of positions are marked when the value has assumed its maximum level.
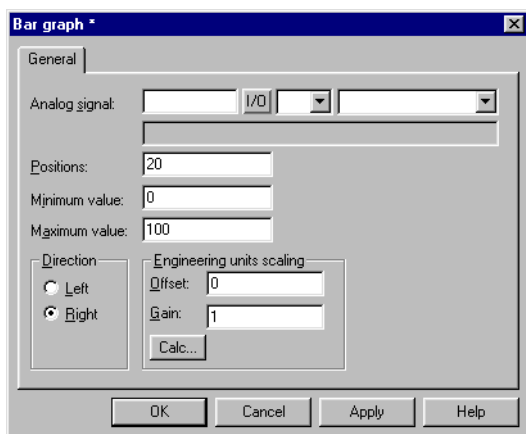
Parameter	Description
Analog signal	Address of the signal.
Positions	The number of positions the signal uses.
Direction	Indicates whether the bar is to be presented to the right or left.
Min value	The minimum value for the object.
Max value	The maximum value for the object.
Engineering units scaling	Used to scale register value. See section *General parameters*.

ABC   **T**ext

Object to handle ASCII strings. It is possible to present texts stored in the controller's data register. The texts should be IBM extended ASCII format. The texts can be changed during run mode from the terminal by opening and closing the input field with [ENTER].
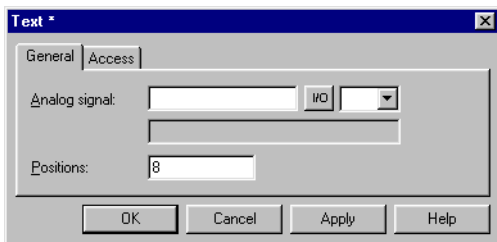
If SW is indicated in the command line in **System signals**, the text is converted from 8 bit IBM extended ASCII to Swedish 7 bit ASCII.

Parameter	Description
Analog signal	The register where the text for the first position is stored.
Positions	Each register contains two characters (assumes two positions). The default setting is 8 positions, which corresponds to four registers.

Access

The functions under Access are described in section *General parameters*.

Jump

Object which performs jump to another block. Jump objects allow menu hierarchies to be built up in the application. To return to the previous block press the [PREV] key (up to nine levels backwards), see chapter *"Function keys"*
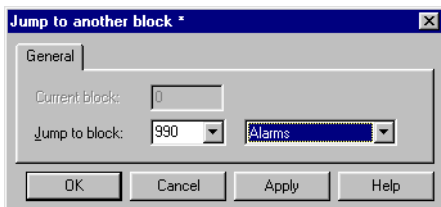
Parameter	Description
Current block	Indicates the number of the current block.
Jump to block	Indicate the number or name of the block to which the jump is to be made.

NOTE   ▮ An error message is received if a jump to a block that does not exist is made during run mode.

MITSUBISHI ELECTRIC

8:05 Date and time

Object for setting the date and time. The date and time objects collect data from the real-time clock in the terminal or the controller.

Parameter	Description
Day of the week	Select whether the weekday is to be shown. Requires two positions.
Date	Select whether the date is to be shown. Use the format indicated under the Date/time format in the configuration.
Time	Select whether the time is to be shown. Use the format indicated under the Date/time format in the configuration.

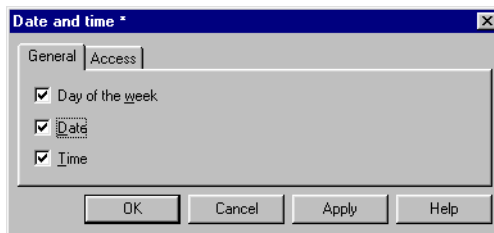**NOTE**   ∎ To set the clock in run mode a maneuvrable Date and time object must be defined.

Access

The functions under Access are described in section *General parameters*.

Message

Object which presents texts from message libraries.
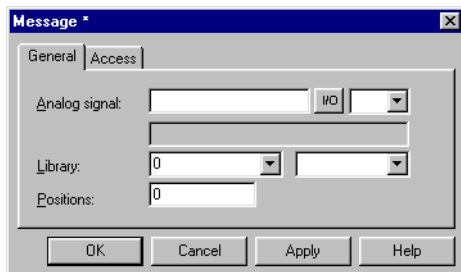
Parameter	Description
Analog signal	Analog signal which controls which text selected from message libraries is to be shown.
Library	Select the number or name of the required message library. Defined under **Functions/Message library**.
Positions	The number of positions the text is to be presented with. 0 = automatic adjusting.

NOTE   ∎ The number of positions should not be 0 when the function indexed message library is used, otherwise automatic length adjusting will not work.

For further information, see chapter *Message library*.

Access

The functions under Access are described in section *"General parameters"*.

6.3   Maneuvre text block

The text block consists of text lines with static and dynamic objects. The dynamic objects show the current status of the signals the objects are linked to. Certain dynamic objects are maneuvrable, which means that you can change their status and value during run mode.

To change a maneuvrable object you use the arrow keys to move the cursor to the correct position so that the object can be marked. Text blocks can be scrolled vertically, but not horizontally.

NOTE   **I**   Objects in text blocks CANNOT be maneuvred in terminals with touch screens.

6.3.1   Digital object

Digital objects are maneuvred by marking the required object. Press [ENTER] to change the status of the object.

6.3.2   Analog object

Analog objects and Date and time objects

The objects are maneuvred by positioning the cursor on the right object. The new value is then entered. Finish with [ENTER]. Before [ENTER] is pressed it is possible to abort the new value by leaving the field with [↑] or [↓]. The old value will then be retained.

Text object

Maneuvre a text object by stepping to the object and pressing [ENTER], whereby an input field is shown. Depending on where the object is placed on the display, the input field will either be on the first or last line. If the text is longer than the width of the display it is possible to scroll in the input field. Press [ENTER] to complete the entry.

Message object,

Maneuvre a message object by stepping to the object with the arrow keys and pressing [ENTER]. A list of options will now be shown with all the maneuvrable states on the display. Step to the required state, press [ENTER] and the analog signal linked to the object will be set.

Multiple choice object

Maneuvre a multiple choice object by stepping to the object with the arrow keys and pressing [ENTER]. A list of options will now be shown with all the maneuvrable states on the display. Step to the required state, press [ENTER] and the analog signal linked to the object will be set.

Jump object,

Mark the object and press [ENTER].

Bargraph

Not applicable for E50. The indicators for min and max values can be set to the current value for the bar object by first marking the object and then pressing [ENTER].

⬧ MITSUBISHI ELECTRIC

7   Message library

Not applicable for E50.

With the function message library you can create text tables, where values between 0 and 65,535 are linked to texts. The message library function can be used for presentation of each sequence step in sequence control. Another possible application is to present error codes. An analog signal generates error codes connected to texts in a text library. This function can also be used to give analog values specific values, depending on the selected texts.

The message library is one of several text tables, each containing up to 512 text strings. Each text string can be 40 characters long. Activate under **Functions/Message library**.
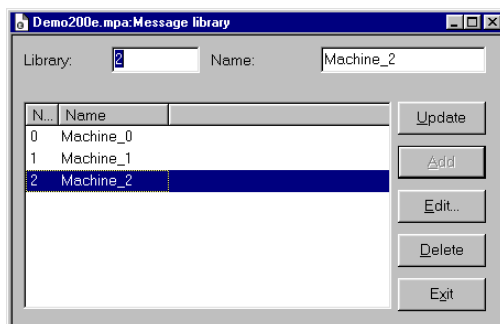
Parameter	Description
Library	Specify a number for the message library.
Name	Specify a name for the message library.

A message library is edited by selecting a library and clicking Edit. Several editing windows can be displayed at the same time.
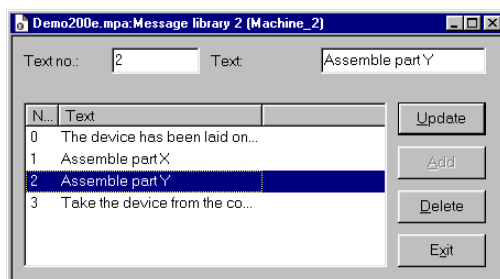
Parameter	Description
Text no.	Specify a number for the text, 0 – 65,535.
Text	The text presented when the current signal assumes the text number for the text.

Example   The following example illustrates how the function can be used. We have a sequence control where each step in the sequence is to be presented by a text.
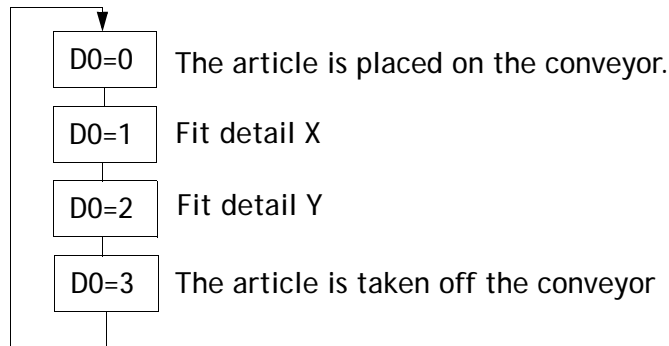
Start by creating a message library called Machine2.

① Select **Functions/Message Library**.

② Specify a number for the library. In this case 2. And a name for the library. In this case Machine2.

③ Click on **Add**.
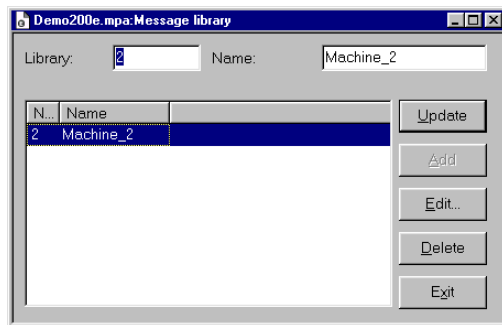
You have now created a message library called Machine2. You should now define the different texts in the library.

④ Select the library and click on **Edit**.

In this dialog you define Text number and Text. **Text number:** is the value of the analog signal linked to the Message object, and **Name** is the text shown in the Message object.

MITSUBISHI ELECTRIC
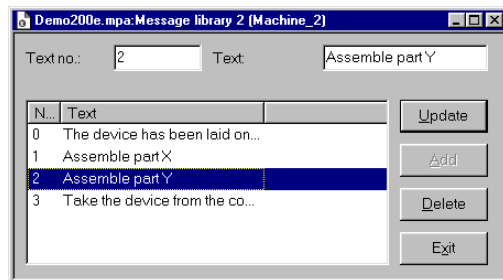
When the message library is ready you create a message object in the application.

⑤ Select the object **Message** in the toolbox.

⑥ Specify the analog signal which is to control which text is to be displayed.

⑦ In the field **Library**, you select the message library from which the texts are to be collected.

⑧ Select whether the object is to be maneuvrable, and between which texts it can be switched during maneuvring in the **Access** tab.

MITSUBISHI ELECTRIC
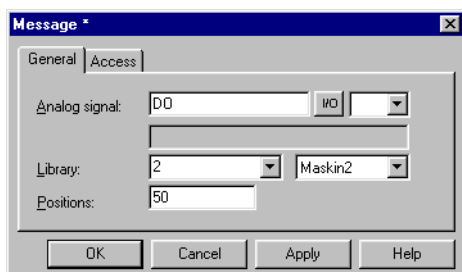
8 Alarm handling

Only applicable for E200.

This chapter describes the alarm handling function. The function is used to call the attention of the operator to events in the process requiring immediate action. It is divided into alarm groups, alarm messages, and an alarm list. The alarms can be divided into groups to create an order of priority. An alarm message determines which signal is to activate an alarm and the text to be shown when the signal is activated. The alarm list contains alarms activated during run mode.

Alarm list
18.2.97 20:11:40 Conveyor 2 empty
18.2.97 20:35:09 Queue full
19.2.97 03:29:12 Low pressure
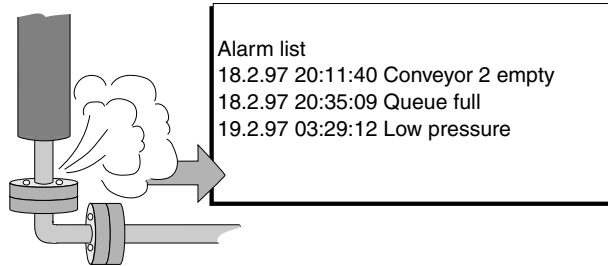
8.1   Alarm message

Alarm messages are defined under Functions/Alarms. The alarm message is entered with a maximum of 18 characters. More characters can be shown if you select a small character size in the alarm list. The message can contain digital or analog dynamic data (as in the case of text blocks), which can be digital or analog. The alarm text can show data from the objects Analog numeric and Digital text. The toolbox is shown when the cursor is in the input field Alarm text, from where you can select to add an object.

You can define up to 300 alarms, depending on the application.

Parameter	Description
Alarm text	An optional alarm text (can also include certain dynamic objects).
Signal	Specify the signal (digital or analog) which when it assumes the specified status generates the alarm.
Alarm when	**Digital signal is**: **On/Off** - Select On if a signal set to one is to generate an alarm, and select Off if a reset signal is to generate an alarm. **Analog signal is**: **Equal to**- An alarm is generated when the value of the given analog alarm signal is equal to the value specified in the subsequent field. **Not equal to**- An alarm signal is generated when the given value of the analog alarm signal is not equal to the value specified in the subsequent field. **Less than** - An alarm signal is generated when the given value of the analog alarm signal is less than the value specified in the subsequent field. **Greater than** - An alarm signal is generated when the given value of the analog alarm signal is greater than the value specified in the subsequent field.
Acknowledge notify	Digital signal activated when the alarm is acknowledged. The signal is normally set to one.
Reset	If the box is marked Reset, the above signal will be reset when the alarm is acknowledged.
Remote acknowledge	Digital signal which acknowledges the alarm when it is activated.
Info block	A block number or block name for a text block is given here. This makes it possible to give the operator a "help page" with, for example, information on the alarm and requisite procedures. If the field is left empty this means that no block is linked to the alarm. If the information block is a text block, it will be included if the alarm is sent as an e-mail. See section *Alarms in run mode* and SMTP client.
Ack. required	Indicates whether the alarm is to be acknowledged or not. A tick in the box means that the alarm must be acknowledged, and no tick means that the alarm functions as an event alarm, e.g. information.

Parameter	Description
History	Indicates when the alarm is to be taken out of the alarm list. A tick means that the alarm will remain in the list until the list is full. No tick means that the alarm will be taken out of the list when it is acknowledged and no longer active. If the parameter **Acknowledge required** is not marked, the alarm will be taken out of the list as soon as it is no longer active.
To printer	Whether the alarm message is to be printed out directly on the printer if the alarm status changes is indicated here.
Repeat count	A counter for the alarm in the alarm list will be shown if the box is selected, which counts the number of times the alarm is generated. The alarm must be acknowledged for it to be presented as a new alarm message in the alarm list.
Import	See section *Alarm import*.

NOTE   ❚ The value defined for an analog alarm signal cannot be controlled from a register. Hysteresis is not supported. Only 16-bit values are supported.

8.2   Alarm settings

In **Setup/Alarm settings** the general settings for the alarms and alarm lists are made. Alarms take up different amounts of space in the alarm list, depending on the length of the alarm text and the number of objects included. The space an alarm takes up can be calculated with the following dialog.

S=42 + NC

S = Number of bytes.
NC = The number of characters in the alarm text.
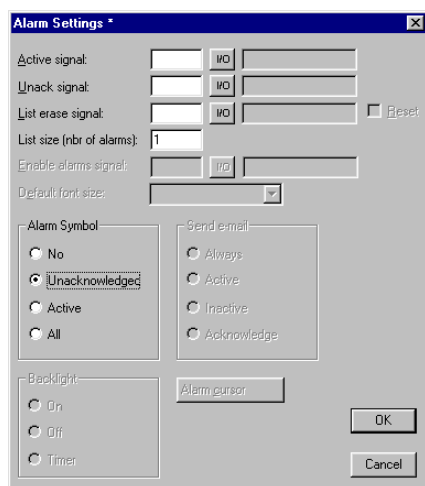
The alarm list is re-written when it is full. 25% is deleted, so that only 75% remains.

Parameter	Description
Active signal	Indicates the digital signal the terminal sets to one when there is an active alarm.
Unack. signal	Indicates the digital signal the terminal sets to one when there is an unacknowledged alarm.
List erase signal	Indicates a digital signal which when set to one deletes non-active alarms in the list.
Reset	Means that the signal given on the List erase signal list is reset when the alarm list is deleted.
List size	Indicates the list size in the number of alarms. Note that when the system allocates memory equal to that which is specified in List size, the actual size of the alarm list is double the indicated size. The performance of the project is adversely affected if the list size is over 100 alarms.
Enable alarms signal	Indicates a digital signal which when set to one activates alarm handling in the terminal. This parameter makes it possible to switch on/off alarm handling in the terminal. The parameter should not be used if alarm handling is to be active all the time.
Default font size	Indicates the default font size in the alarm list. The default font size in the alarm list is always shown after restart or start-up, and after switching between operation modes.
Alarm Symbol	Indicates when the alarm symbol is to be shown. In text blocks ■ALARM■ is shown in the top right-hand corner of the display. **No** means that the alarm symbol is never shown. **Unacknowledged** means that the alarm symbol is shown when there are unacknowledged alarms in the alarm list. **Active** means that the alarm symbol is shown when there are active alarms in the alarm list. **All** means that the alarm symbol is shown when there are active and/or unacknowledged alarms in the alarm list.

🔶 MITSUBISHI ELECTRIC

8.3   Alarm import

It is possible to import alarm texts from name lists (generated from the programming tool for the controller). The project in the programming tool must be linked to a name list before alarm importing can be carried out. Select the relevant name file under **View/Namelist**. Now select **Functions/Alarms** and click on the button **Import** to define the alarm import.

Start I/O

Indicate the address for the first I/O of the import from the name file. Can be analog or digital signal.

End I/O

Indicate the address for the last I/O for the import from the name file. Can be analog or digital signal, but must be of the same type as Start I/O

Alarm settings

All alarms imported (Start I/O to End I/O) when you click on the button **Import** will have the same settings as the settings defined in the box **Alarm Settings** in the dialog (Alarm Import). For a description of the parameters, refer to section *Alarm message*.

The parameters, signal type, analog or digital and alarm group must be specified before an alarm import can be made.

8.4   Alarms in run mode

In text blocks, alarms are indicated by showing the text ■ALARM■ in the upper right corner of the display. In graphic blocks a bell flashes in the upper-right corner of the display when an alarm is activated. Indication can be deselected in Configuration mode or under **Setup/Alarm settings** in the programming tool.

Alarms are presented in an alarm list with predefined alarm texts. The alarm list contains the latest alarms and is arranged in alarm group order according to definition, so that the latest alarms are shown at the top of the list. The size of the alarm list in the number of kbytes (number of alarms in E200) is determined in configuration mode or under **Setup/Alarm Settings** in the programming tool. If there is a jump to an alarm block (system block no. 990), the alarm list will be shown.

The number of times the alarm has been generated (if selected), the status of the alarm, the time it was activated, became inactive or was acknowledged, is shown for every alarm, depending on the chosen display format.

The counter for the alarm in the alarm list, which is shown if the box **Repeat** count is checked for the alarm in the Alarm dialog, is shown as follows.

Display format	Explanation
(12)	Means that the alarm has been generated 12 times. The alarm must be acknowledged for it to be presented as a new alarm message in the alarm list.
>999)	Means that the alarm has been generated more than 999 times without being acknowledged between each time. The counter counts to a maximum of 999.

Alarms can have the following status.

Symbol	Status
*	Active, Unacknowledged
$	Active, Unacknowledged
-	Active, Acknowledged
<blank>	Inactive, Acknowledged

Alarm times can be shown in the following format.

Display format	Explanation
S	The time when the alarm was activated. The time the alarm was first activated is shown for repeated alarms.
E	The time when the alarm was inactivated. The time the alarm was last inactive is shown for repeated alarms.
A	The time when the alarm was acknowledged.

To access the alarm block you can either define a jump to system block 990 in a block, press [LIST] or allow the controller to present the list through Display signal for block 990.

To acknowledge an alarm place the cursor on the line where the alarm is and click [ACK], point at the symbol ✓ or acknowledge with a function key. If a printer is connected the alarms can be printed out as they occur, or change status. In which case this is specified when the alarm is defined. Alarms will be printed out with the number of events, date, time, status, and alarm text.

Historical inactive acknowledged alarms can be deleted by pressing the [←] (BACK SPACE). By entering the command DD on the command line under **Settings/System Signals** the function will be disabled.

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

By selecting Print signal for block 990 it is possible to print the current content in the alarm list.

8.4.1   Block linked to alarm

Text blocks can be linked to alarms. This means that when the operator presses the INFO button for an alarm in the list, the block linked to the alarm will be shown. This block can contain information on the alarm and suggestions for possible procedures. It is only possible to press the INFO key if the currently selected alarm is linked to a block. Return to the alarm list by pressing [PREV].

MITSUBISHI ELECTRIC
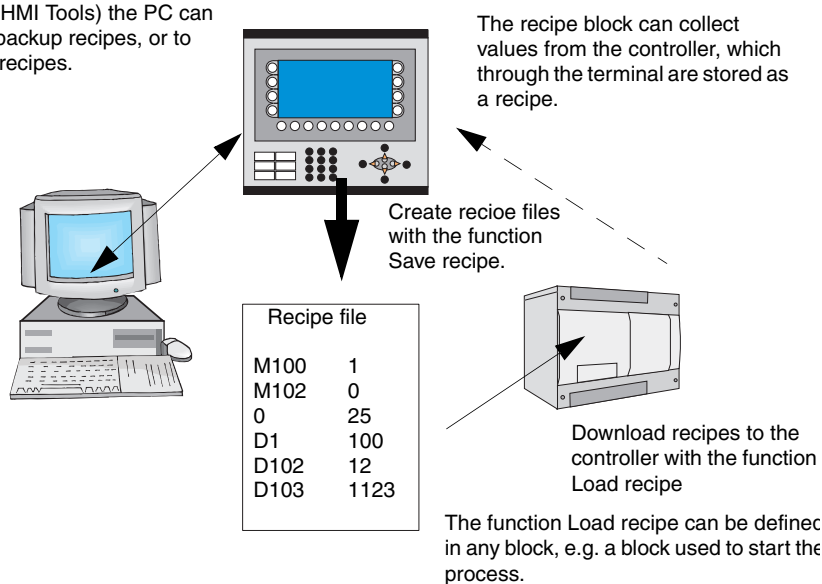
9 Recipe handling

Not applicable for E50.

This chapter describes the function recipe handling. The function Recipe handling means that all the dynamic data in one or more blocks, i.e. the signals and their values, can be saved to file in run mode. The operator can then download the files to the controller, which will start working with the new values. By using the function for recipe handling it is possible to reuse large sets of parameters. The user can create recipe libraries consisting of files containing different sets of parameters. This function improves the efficiency of time-critical productions where the exhange of similar products must be done quickly, such as in the manufacturing of units in different colors. Recipe files can be created from the terminal, the controller, or from a PC with the program HMI Tools.

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

The figure below illustrates the principle of how recipehandling functions.

The PC is used to program the terminal. With the program File Transfer (in HMI Tools) the PC can be used to backup recipes, or to create new recipes.

The recipe block can collect values from the controller, which through the terminal are stored as a recipe.

Create recioe files with the function Save recipe.

Recipe file	
M100	1
M102	0
0	25
D1	100
D102	12
D103	1123

Download recipes to the controller with the function Load recipe

The function Load recipe can be defined in any block, e.g. a block used to start the process.

9.1   Calculation of recipe size

To calculate how much space the required recipe will take up in the project memory, the following equation is used. (The equation is not exact in view of the file system's complexity in the terminal.)

$$S = 90 + \Sigma \, (2 * IOG_i + 28)$$

S = number of bytes. If S is calculated to be less than 360, then S is set to 360.

Σ = number of I/O series.

IOG_i = number of I/O in each I/O series

Example   We have a recipe consisting of 3 I/O series D0-D109 (110), D200-D499 (300) and M0-M99 (100).

The calculation will then be:

Calculate the subtotals of the different series:

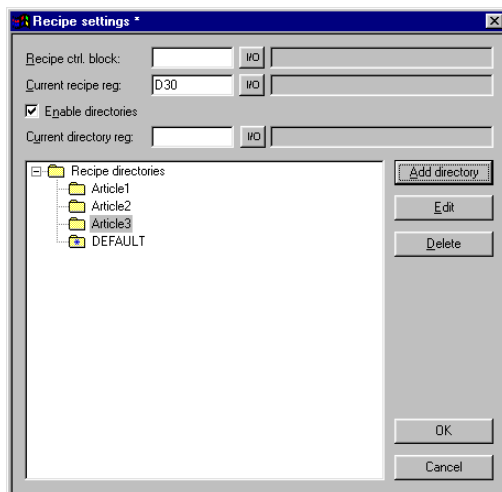$$1104 = (2 * 110 + 28) + (2 * 300 + 28) + (2 * 100 + 28)$$

Then calculate the total:

S = 90+ 1104

S = 1194 bytes

9.2   Recipe settings and recipe directory

Under **Setup/Recipe Settings** you enter settings for recipe handling and create, edit and delete recipe directories.

Recipe ctrl. block

Control block to save/read/delete recipes through the controller. See section *Creating and transferring recipes with the controller program*.

Current recipe reg:

You specify here the first of four 16 bit registers where the terminal inserts the name of the last loaded recipe to the controller. The name can then be presented in an ASCII object. The function always uses four registers (eight characters) irrespective of the length of the recipe name.

MITSUBISHI ELECTRIC

Enable directories

If the box is selected you can create a recipe directory in the terminal. See section *Recipe directory*.

Current directory reg:

You specify here the first of four 16 bit registers where the terminal inserts the name of the recipe directory specified for the block. The name can then be presented in an ASCII object. The function always uses four registers (eight characters) irrespective of the length of the recipe directory. See section *Recipe directory*.

9.2.1   Recipe directory

Recipes created in the terminal can be saved in different recipe directories (folders) in the terminal's memory. The use of recipe directories ensures a better structure and more simple handling of recipes in applications with a large number of recipes. It is possible to create 8 different recipe directories in one level. Recipe directories are saved under the directory RECIPE in the root directory in the terminal's memory. A recipe directory is linked to one or more blocks in the block header for the block. All recipes created in the block will then be saved in the selected recipe directory.

Recipe directories are created, edited and deleted under **Setup/Recipe Settings** in the programming tool. Defined recipe directories are shown in a list, according to the structure in the directory. Add new recipe directories with the button **Add directory**. Recipe directories must be named with a name having 1-8 characters. The first character must be a letter or digit, and the name must contain the characters A-Z, 0-9 and _. Otherwise, the standard for file names in MSDOS is followed.

Change a recipe directory by selecting it and clicking **Edit**. **Delete** deletes selected recipe directories.

Recipe directory in run mode

In run mode recipe directories can be created and deleted with the functions **Make recipe directory** and **Delete recipe directory**. The functions are linked to function keys. With the function **Change recipe directory** for function keys it is possible in run mode to change/select recipe directories for the current block. A list of options for available recipe directories is shown when the function key for Change recipe directory is pressed. Select one and press [ENTER]. The recipe in the block will now be saved in the selected recipe directory. See section *Function keys*.

NOTE   **I**   Recipe directories created in the programming tool cannot be deleted with a function key linked to the function Delete recipe directory. Recipe directories created in the terminal do not accompany the terminal project when a project is uploaded from the terminal to the programming tool (The function Receive in the dialog project transfer).

Recipe handling between terminal and PC is performed with the program HMI Tools File Transfer and HMI Tools FTP Transfer. See section *Using recipes in a PC* and the manual for HMI Tools.

9.3   Creating a recipe with the terminal

When the application is programmed you determine which block or blocks are to be used to save recipes, or alternatively the function **Append recipe** is used in run mode. All the signals to be included in the recipe are defined in the recipe block. All the dynamic values in the block are stored in the recipe file. All digital and analog objects can be used as recipe parameters.

In run mode a jump is made to the block containing the recipe parameters. Enter the required values in the dynamic objects and press the function key linked to **Save recipe**. The name must have 1-8 characters. The first character must be a letter or digit, and the name must contain the characters A-Z, 0-9 and _. Otherwise, the standard for file names in MS-DOS is followed. Recipe files are stored in the terminal, either in the recipe directory specified for the block or in one and the same recipe directory if no separate recipe directories have been created under **Setup/ Recipe Settings**.
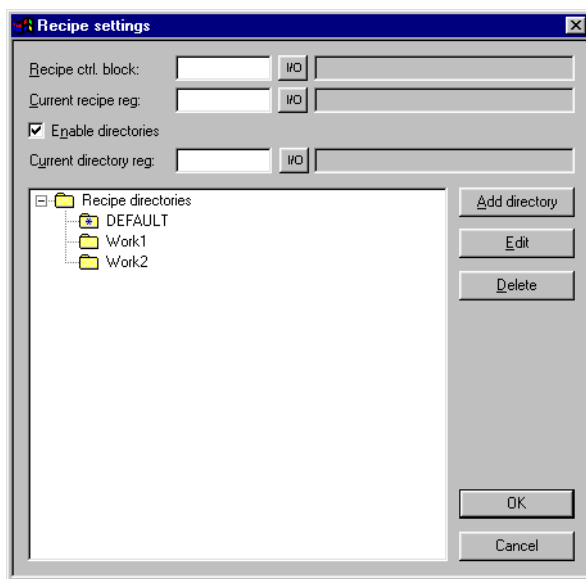
9.4   Append recipe

The function **Append recipe** can be linked to function keys and makes it possible in run mode to add signals and their values from the current block to an existing recipe. I.e. the operator can save signals and their values from several different blocks in a joint recipe. New signals are added to a recipe, and signals already existing in the recipe will be updated when the function is activated. The name of the recipe to which the current block's signals and their values are to be added is specified when the function key for Append recipe is pressed. If the function is activated and there are no recipes saved in the terminal, a new recipe will be created in the terminal. Similarly, a new recipe will be created if the same recipe directory for the block is not given. To add signals from another block to a recipe the same, or no recipe directory, must therefore be given for both blocks.

NOTE   ❙ If a string is added to an existing recipe with strings the strings must be separated by address space. Otherwise the previous near string will be increased.

✶ MITSUBISHI ELECTRIC

9.5   Transferring recipes to the controller

In run mode recipes are transferred to the controller with the function **Load recipe**. This function means that the signals and their values stored in the file are transferred to the controller. A list of options for available recipe directories is shown when the function key for **Load recipe** is pressed. Select one, and press [ENTER]. The controller will then run with the downloaded values.
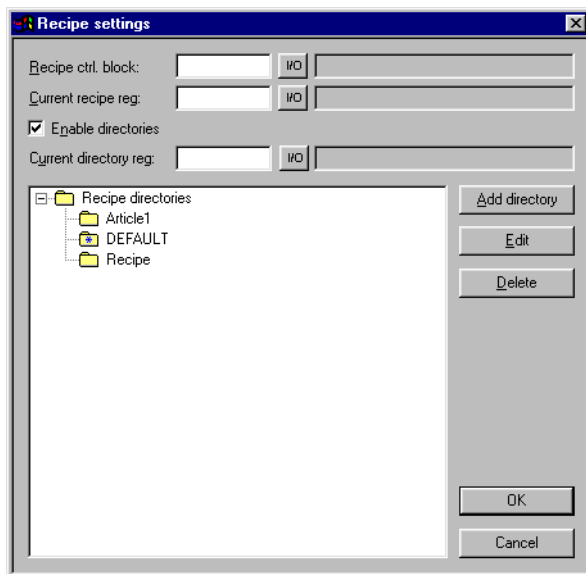
9.6   Delete recipe

In run mode the selected recipe can be deleted from the terminal memory with the function **Delete recipe**. Press the function key linked to **Delete recipe**. A list of options will now be shown with the available recipe files. Select the file to be deleted, and press [ENTER]. Confirm with [ENTER] or press [PREV] to abort.

9.7   Using recipes in a PC

With the program HMI Tools File Transfer (icon in the program group HMI Tools) in the PC, the terminal will be treated as a unit in the PC. This means that the PC can be used to make backup copies of files in the terminal, e.g. recipe files. New recipes can be created in the PC and transferred to the terminal.

Recipe files are stored in the PC in .SKV format, which can be brought up in Excel. The files can be edited in Excel and then used in the plant again. The files must end with the instruction END. See the example below.

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

It is also possible to transfer recipe files between the terminal and the PC through FTP with the program HMI Tools FTP Transfer (standard FTP client). For further information, refer to the manual for HMI Tools.

9.8   Creating and transferring recipes with the controller program

Through a control block in the controller you can create, transfer and delete recipe files in the terminal. The files created from the controller program are compatible with the recipe files created from the terminal. I.e. recipe files created in the controller program can be downloaded into the terminal, and vice versa. The control block appears as follows.
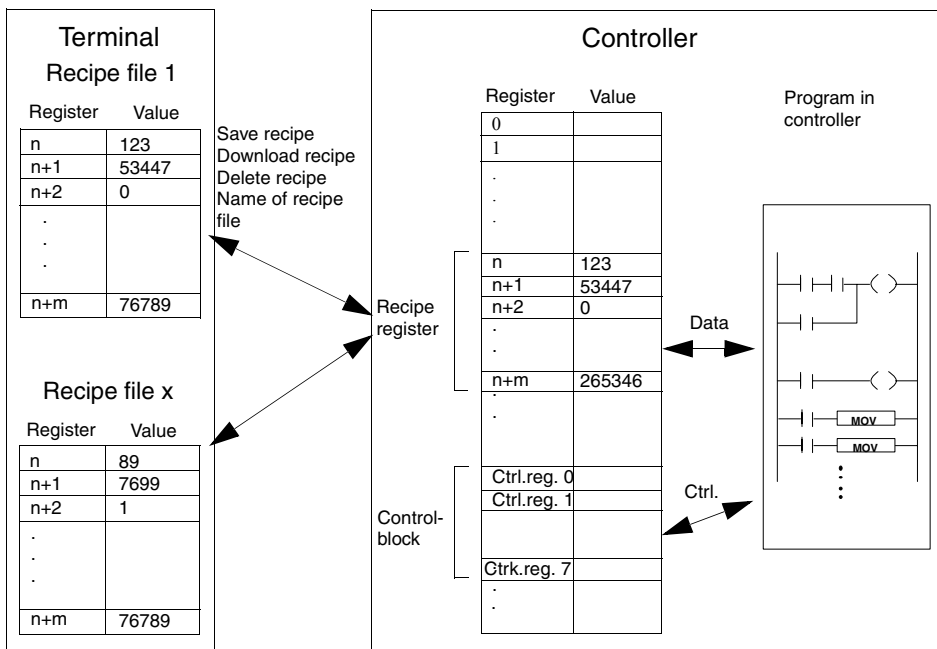
Under **Setup/Recipe settings** you define the first register in the control block. This and the seven following registers are used as control registers. The control block is described in the following table.

Register	Content	Description
Ctrl reg 0	Command	Command register set by the controller. Available commands: 0: No command. 1. Save recipe to terminal. 2. Read recipe from terminal. 3. Delete recipe from terminal. 4. Create recipe directory. 5. Change recipe directory. 6. Delete recipe directory.
Ctrl reg 1	Handshake	Handshake register set by terminal. 0. Ready for new command. 1. OK 2. Recipe file write error. 3. Recipe file not found.
Ctrl reg 2	File name char. 1-2	
Ctrl reg 3	File name char. 3-4	
Ctrl reg 4	File name char. 5-6	Name of recipe file/recipe directory in the terminal.
Ctrl reg 5	File name char. 7-8	
Ctrl reg 6	Start data register	The first data register to be read/written to/from recipe file.
Ctrl reg 7	Number of registers	Number of registers to be read/written to/from recipe file.

MITSUBISHI ELECTRIC

The following procedure must be carried out:
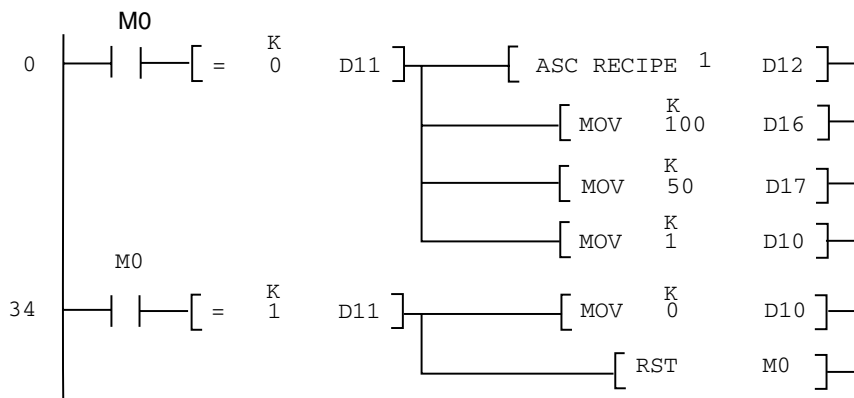
⑨ The handshake register must be 0. If not, reset the command register to 0.

⑩ Enter the command in the command register.

⑪ Wait for the ready signal or error code in the handshake register.

⑫ Set the command register 0. The terminal will now set the handshake register to 0.

NOTE   ▮ Recipe directories created in the programming tool cannot be deleted in run mode. See also section *Recipe directory in run mode*.

Example   When memory cell 0 is activated the controller will save data register 100-149 as a recipe with the name RECIPE1. The control block starts on data register 10. The controller program should now be as follows:

```
         M0
                        K
  0    ┤├──────[ =     0     D11 ]────────[ ASC RECIPE  1     D12 ]──┐
                                                         K
                                          ─────[ MOV    100   D16 ]──┤
                                                         K
                                          ─────[ MOV    50    D17 ]──┤
                                                         K
                                          ─────[ MOV    1     D10 ]──┤
         M0
                        K                                K
 34    ┤├──────[ =     1     D11 ]────────[ MOV    0     D10 ]──┐
                                          ─────────[ RST        M0 ]──┤
```

Limitations

Recipes created in the controller can contain a maximum of 1000 registers. Only data registers can be used.

The following characters cannot be used in recipe names:

! ? < > ( ) + / * = × % # : . [space] , and -

10   Password

Not applicable for E50.

This chapter describes the handling of security levels and passwords in the terminal. It is a function which makes it possible to create a security system for the plant. The operators can easily be assigned different authorizations in the plant.

Blocks, function keys and maneuvrable objects can be allocated a security level. Each security level is allocated a password. To access the different levels the user must login with password for the relevant security level, or higher. It is not necessary to use this function.

10.1   Defining security levels

Blocks, function keys and maneuvrable objects can, when they are defined, be allocated a security level. A security level (0-8) is specified in the dialog box for the object under the **Access** tab after the **Enable operator input** has been selected. If security level 0 is given, the object will be available for everyone, i.e. login will not be requested.

10.2   Defining password

Under **Functions/Password** you define passwords for the security levels 1-8.

Parameter	Description
Password 1-8	Enter password for security level 1-8.
Confirm question 1-8	Enter the question, with a maximum of 20 characters, to be answered by the operator before maneuvring of an object with the defined security level is possible. This function will not function if a function key has been defined with the security level.
Comment 1-8	Enter comments/description for the password/security level. This parameter does not need to be used.
Login signal	Specify the digital signal, which when it is set to one generates the input field for logging in. The input field for logging in can also be linked to a function or touch key, see chapter *Function keys*.

Parameter	Description
Logout signal	Specify a signal, which when it is set to one logs out the current user. The function can also be linked to a function or touch key, see chapter *Function keys*.
Login level reg	A register in the controller to control the security level can be specified here. The value in the register determines the current security level, value 0 = no security level, 1 = security level 1, etc.
Current level reg	Specify a register from where the terminal can present the current security level (0-8).
Login timeout	Specify the time in minutes which the terminal can remain inactive before the user is automatically logged out. If 0 is given there will be no logout.
Password RUN/PROG	You can specify a password to be given for manual switching from RUN to PROG mode. The function is not used for switching from PROG to RUN or when Automatic terminal RUN/TRANSFER switching is used from the programming tool.
Automatic login	Specify if the login window is to be automatically shown when trying to access password-protected blocks, objects or keys. The function is only applicable for terminals with touch screens, since it is not possible to place the cursor on password-protected objects without being logged in on the correct security level for the object.

10.3 Logging in

If the box for Automatic login under Functions/Password is not checked the login will either be controlled from a function key, or through a digital signal from the controller (**Login signal**). If the function key linked to the function **Login to specified security level** is pressed, or the digital signal is activated, the input field for login will be shown. Specify password. The password is linked to a security level, see above.

10.4 Password for project transfer

On the command line under **Setup/System Signals** you can enter the command PDxxxxxxxx. The command means that a password (xxxxxxxx) must be given to access the functions in the menu TRANSFER in the terminal. When transferring projects from the programming tool to the terminal this password must be given in the terminal.

10.5 Overriding password

On the command row under Setup/System signals..., the command PSxxxxxxxx can be entered. When the password is entered (xxxxxxxx), the command overrides all other password levels and the user gains access to everything in the terminal. This is useful, for example, in performing support and maintenance. On the command line, only capital letters may be specified.

10.6 Change password during run mode

The password can be changed during run mode with the function **Change login password** for function keys. When function keys linked to Change login password are pressed a dialog box is shown, where the password for respective security levels can be changed. See chapter *Function keys.*

NOTES   ∎ It is not possible to select a security level on block 0.

∎ After logging out, the [PREV] key and function **Return to previous block are ignored for function keys to p**revent unauthorized persons accessing block protected by passwords.
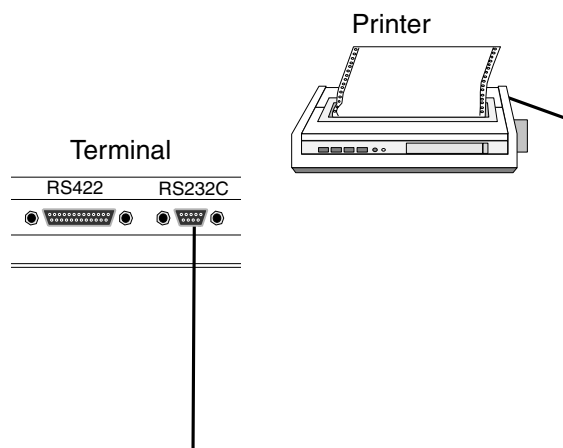
11   Printouts

Not applicable for E50.

This chapter describes how report printouts are made in the terminal. Different types of reports, e.g. daily reports and event reports, can easily be created for production follow-up. The following diagram shows the principle for generating daily reports.
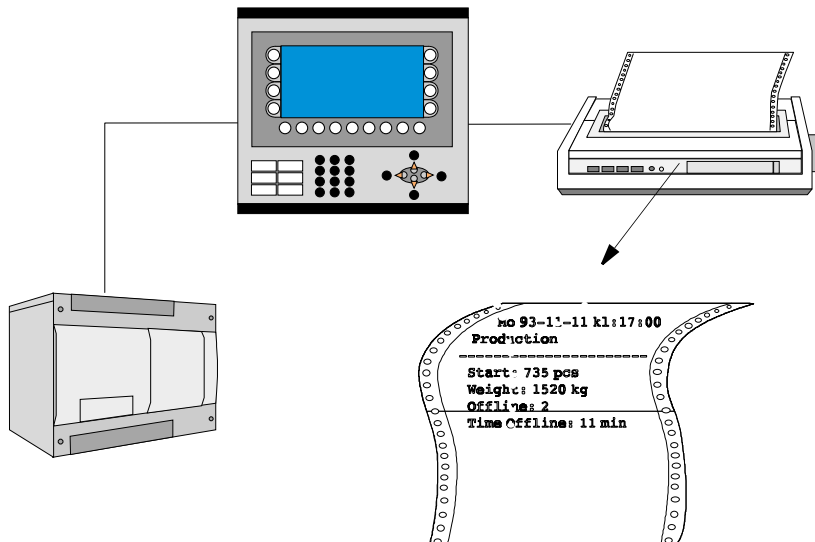
11.1   Connection to printer

The printer should have a serial interface and an IBM character set (850). The settings for the printer are made in the dialog under **Setup/Peripherals**. For configuration of the printer, refer to the printer manual.

NOTE   ▌ The expansion card IFC PI must be installed in the terminal to connect a printer with parallel interface to the terminal.

Examples of printer:

Serial printer   Panasonic KX-P1092

Parallel printer   HP Laserjet GL
                   HP Laserjet 1100

11.2   Project printouts

Choose **File/Print...** to print the project.
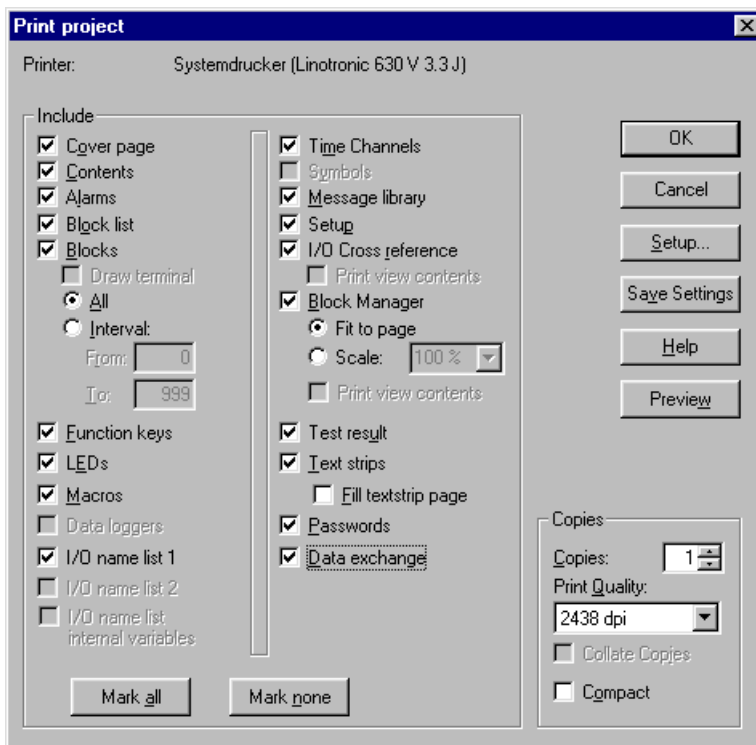
Choose the portions of the project to be printed by clicking the boxes. Click **Setup** to configure the printer. Click **Preview** to view the printout before sending it to the printer.

11.3   Text block reports

Reports are created as text blocks with static and dynamic text. The width of the reports is a maximum of 150 characters. In the text block you write an optional text, e.g. a table header, and other static text which is always printed. To print the values from the process you define the dynamic object which presents the value for the signal the object is linked to.

See chapter *Text-based presentation and maneuvring*. You can decide when the report is to be printed through time channels, for example.

11.4   Defining the printout

The printout is defined in the block header. You obtain the block header from the Block Manager or the block list. The parameter **Print signal** in the block header indicates the digital signal, which when it is set to one activates the printout of the block. A digital **Completion signal** is also indicated here, which is set to one by the terminal when the printout is complete. If the box **Reset** is selected, the signal will be reset instead.

11.5   Printer settings

The printer settings are made under **Setup/Peripherals/Printer/Edit...**. For more detailed information concerning printers, refer to the manuals for respective printers. The printer musts support an IBM expanded ASCII character set.
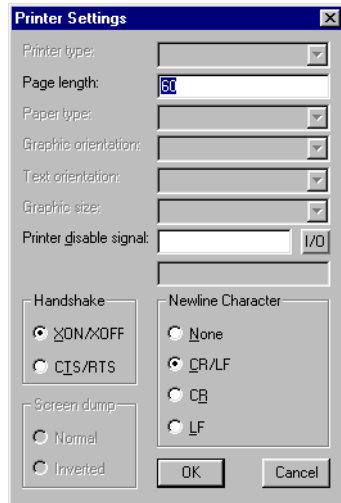
Parameter	Description
Page length	The number of lines to be printed out before form feed. If the page length is set 0 there will be no form feed.
Printer disable signal	Indicates a digital signal, which when it is set to one, stops the printout. The port where the printer is connected can then be used for other communication, e.g. Transparent mode. Not applicable if IFC PI is used.
Newline character	Line end character; CR/LF, CR, LF or none.
Handshake	Indicates whether handshaking between printer and terminal is to be with XON/XOFF or CTS/RTS.

11.6   Setting of communication port

The settings for the communication port are made under **Setup/Peripherals/ RS-232C** or
RS-422. When you mark **RS-232C** or **RS-422**, click **Edit...**. See the manual for the connected
printer for the correct settings.
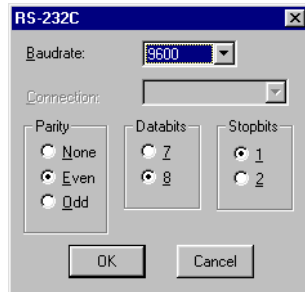
Parameter	Description
Baud rate	Select the baudrate for the communication; it should be the same as for the external unit.
Parity	Select the parity; it should be the same as for the external unit.
Data bits	Select the number of data bits; it should be the same as for the external unit.
Stop bits	Select the number of stop bits; it should be the same as for the external unit.

11.7   Printer status

It is possible to read the status of a printer connected via the expansion card for parallel printer
by using a printer status register. The register is selected under **Setup/System signals.** For a
description of the printer status register, refer to the section **System Signals** in chapter *Pro-
gramming with the programming tool*.

MITSUBISHI ELECTRIC

12   Time control

Not applicable for E50.

Time control is a function for setting and resetting digital signals in relation to the real-time clock. This function is used to control events in the process at special times, e.g. starting and stopping motors, through the terminal. The time channels replace the time relays and 7-day clock.

12.1   Defining time channels

Time channels are defined under **Functions/Time channels**.
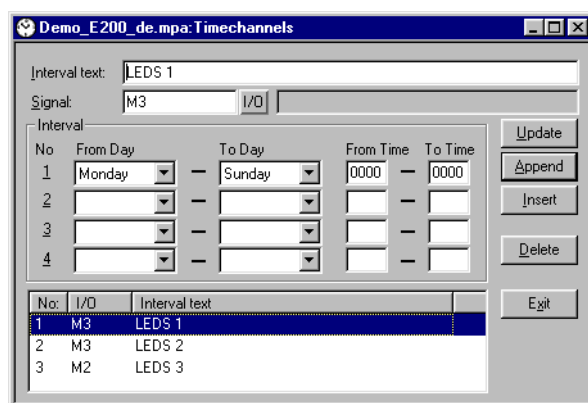
Parameter	Description
Interval text	An optional text is entered here for the time channel.
Signal	A digital signal is specified here, which is set to one during the specified time interval.
Interval	The days and times for the interval are specified here. Four different intervals can be defined for each time channel.

In the picture above we have defined one time channel. This time channel has the text Night temperature, and is linked to the output signal 30. The signal will be set to one Monday to Friday between 17.30 and 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 reduced.

12.2   Presentation in run mode

The page with the time channels is shown when system block 991 is activated, either through a jump object or through a digital signal linked to the block. The values of the time channels can be read and/or changed. To change the value of the time channels in run mode, the alternative **Time channels** under **Setup/Online settings** must be marked.

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

▲ **MITSUBISHI ELECTRIC**

13   Language management

Does not apply to E50

The programming tool supports several language applications for the E-series' operator termi-
nals. We advise that the entire application be created in one language in the programming tool.
Using the multiple language support function, you can later easily translate all texts in the appli-
cation to other languages. Translation can be performed directly in the programming tool or the
text can exported as a text file and translated in other software. The translated file is later
imported back into the programming tool. A maximum of 10 languages can be created for an
application.

Each text in the application can be allocated an arbitrary index. To optimize the function and
reduce the total number of texts, a text can be used several times in the application, copied and
reused. These texts are linked to the same index.

The application language contains user texts and is linked to a system language that contains
system texts. User texts are those texts that are entered when the project is programmed and
system texts are those texts that already exist when a new project is created as well as texts that
already exist in the terminal's system program.

13.1   Creating additional application languages

Choose **Setup/Multi language.../New language**. This starts the Multiple Languages wizard, a
self-descriptive sequence of dialog boxes for creating additional application languages. Follow
the instructions in the respective dialog boxes and select or type the desired parameter values
or name(s).

Choose to copy all texts from an existing language, i.e. a previously created language.
Language 1 is the language in which the application was created (basic language).
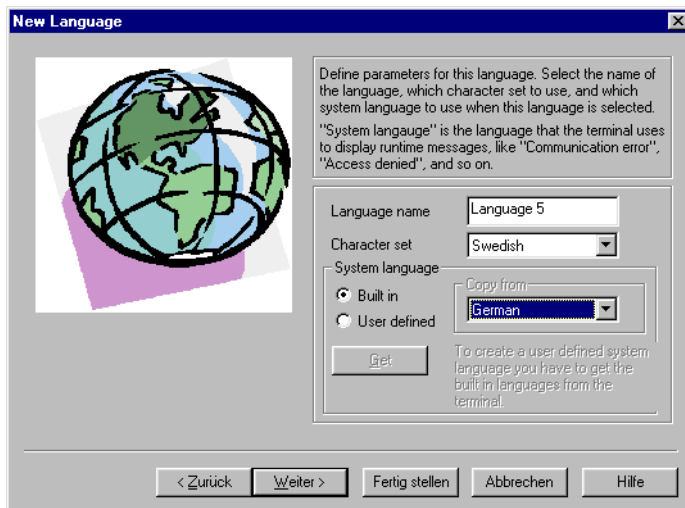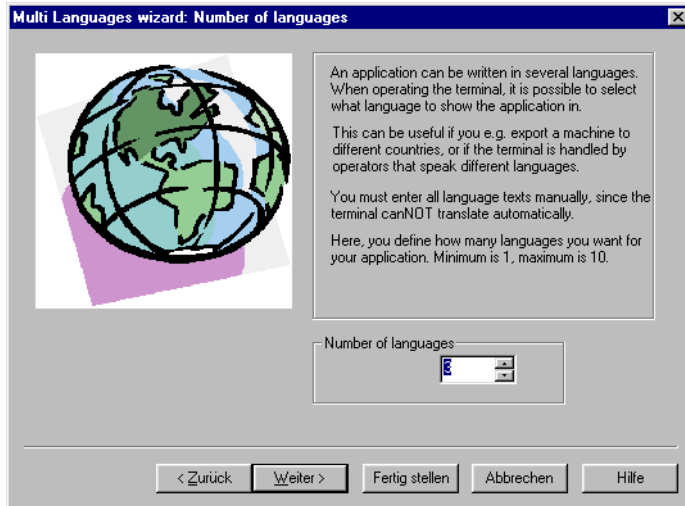
The software suggests language names. To clarify matters for the user, the name should be changed.

Under **Character set**, you choose the character set that will be used on the terminal and the national characters that will be available. See the section *National Settings*.

Under **System language**, you can choose **Built in** or **User defined**. **Built in** entails that the system texts in the terminal are displayed in the chosen language. **User defined** entails that you can translate a built in system language and link the new system language to the application language (requires that the appropriate terminal be connected to the PC in accordance with the chapter *Transfering projects*.)

◢◣ **MITSUBISHI ELECTRIC**
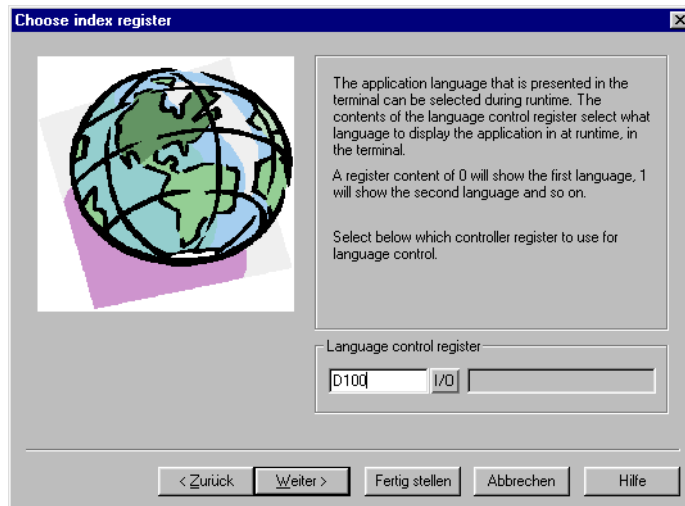
This is where the Language control register is specified. Its value (0-9) during runtime determines the application language (0-9) that the terminal will use.

Click **Finish** to close the wizard. A structure tree is now displayed with the languages you have created.

13.2 Translating/Editing texts in the programming tool

Choose **Setup/Multi language.../Edit**.
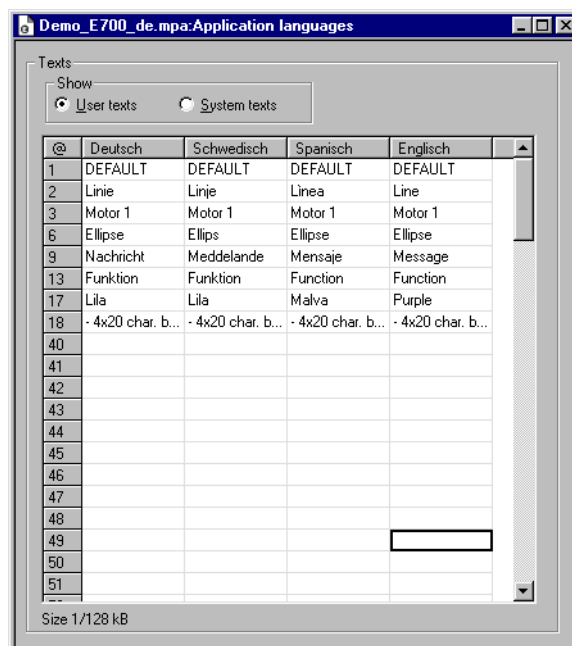
This is where the translations for the various languages are entered in the respective table cells. Move the cursor between the rows and columns with the arrow keys. You can search within the text list with the menu command **Edit/Find**.

An application language can also be exported as a text file and translated in another program, such as Excel or Notepad. The text file is thereafter imported back to the application. See the sections *Export* and *Import*.

13.3   Settings for the application language

Choose **Setup/Multi language.../Setup**.
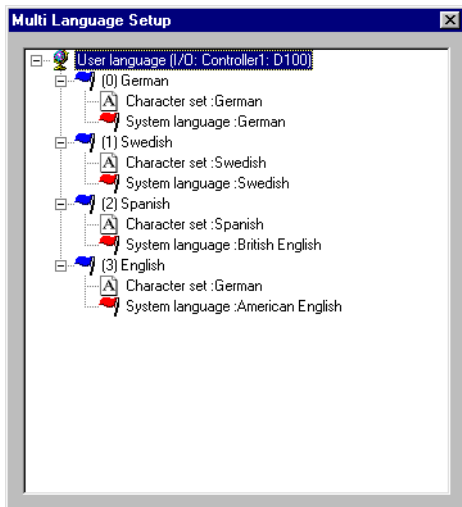
Right-click **User language** to change the register that determines the displayed language.

When you right-click the language name, you can choose between the following menu commands:

Menu command	Description
New language	Creates a copy of the selected language
Character set	Chooses/edits character set
System language	Chooses/edits system language
Delete language	Deletes the selected language
Rename	Changes name of selected language
Properties	Specifies data register with the value that determines the displayed language

Right-click **Character set** to change the character set (also applies to Unicode) for the language.

Right-click **System language** to change the selected system language or create a new one.

13.3.1   User-defined system languages

To create a user-defined system language, choose **User defined**, choose the language that would like to copy from and click **Get**. The **Language Transfer** dialog box opens. Click **Receive** to retrieve the built-in system language from the terminal. System texts are now accessible for editing under **Setup/Multi language.../Edit**. You can also choose to export the texts as a text file and edit them in another program.

All system texts in the terminal (passwords, time channels, etc.) support the multiple language application. You can use the predefined system languages or create your own (new). All types of characters that exist in the character set can be used in the application language. A text string is linked to several different objects. The maximum number of text strings for each language depends on available project memory in the terminal. The following spaces are available for each language:

E100 - E200= 16KB/language

At the bottom left of the **Application Language** dialog box, information is displayed on memory size for selected languages (language file). The information is displayed in the format X/Y. X is the utilized memory and Y is the available memory for each language (for example, 7/128).

13.4   Export

You can choose to export a language to Excel, for example, translate it there, and then import it back to the programming tool.

Choose **Setup/Multi language.../Export**. Choose **User texts** (or System texts). In the displayed dialog box, specify the name of the file to be exported and indicate if it is to be an ANSI, OEM or Unicode text file.

ANSI/OEM entails that languages created in the ANSI/OEM format will be exported, and **Unicode** entails that all languages will be exported to a file in the Unicode format. Editing a Unicode-format file in another application requires that a Unicode font be selected in that application.

13.5   Import

Choose **Setup/Multi language/Import**. Choose **User texts** (or System texts). The **Import Multi language texts** dialog box is displayed. Choose the name of the text file that you would like to import. If the existing language in the project in is in ANSIOEM format and a language in Unicode format is imported, the imported language will be converted to ANSI/OEM.This means that all characters not included in the ANSI/OEM range will be displayed as question marks.

NOTE   ▌ If a project with several application languages is opened in MAC Programmer+ will replace all objects with several application language texts with the character @ followed by an index number (for example, "@55").

13.6   Show index

Each object that displays text in a multiple language application is linked to an index. An index can be linked to different objects with the same texts. Choose **Setup/Multi language/Show index** to display the index number for the object texts.

13.7   Cross reference

Choose **Setup/Multi language/Cross reference**. On the displayed cross-reference list you can right-click an object to edit it. The cross-reference list has support for incremental searching when you specify an index number.

13.8   Copy object

When **Reuse index** is active and an object is copied, the copy will be assigned the same index. This entails that objects with the same index need only be translated once. Changes made to a text will be seen at all locations where the same index number exists.

NOTE   ▌ If an object is deleted that has copies with the same index number, only the selected object is deleted.

13.9   Choose Unicode font

Here you can choose a Unicode font if an increased number of characters is needed to create the application languages.

13.10   Application languages in run mode

You can change the application language in the terminal in run mode by changing the value (0-9) in the specified language selection register. Note that the terminal redraws/rewrites the entire block when a new language is selected in run mode (RUN). If there is a cursor in the terminal,this will be placed at the first maneuverable object in the current block.

14 Indexadressierung

Not applicable for E50.

Without index registering an object is always connected to the same register, and only the value in this register can be shown in the object. With index registering, however, it is possible to select during run mode from which register an object is to collect the shown value. This can be done by adding the value in the index register to the address of the register which is the analog signal in the object. In general this can be formulated as:

Display value = the content in Register (the Object's address + the content in the Index register)

If the content in the Index register is 2 and the address of the register specified in the object is 100, the value shown in the object will be collected from address 102. If the value in the Index register is changed to 3, the value in the object will be collected from address 103 instead.

Which register is to be the index register is defined in the individual project. This is done under **Setup/Index register**. Up to eight different index registers can be used in each project, and each index register can be used for more than one object.

The objects used in the project will specify whether index registering is to be used, and if so, which index register. This is done by selecting in the dialog for the object I1 to I8 in the box to the right of the indicated address signal for the object. In the example below, D100 is the analog signal and I1 is the index register used.
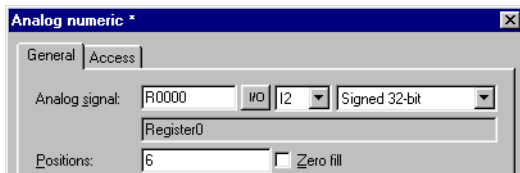
In the following example, three motors are to be controlled from one block. The torque and speed for the three motors are stored in six different registers. One of the motors will be selected in one block, and then only the current torque and speed for this motor will be shown in the block. If another motor is selected, the current torque and speed will be shown for this motor instead. This is possible with index registering.

Motor 1   Motor 2   Motor 3

Torque in register D101   Torque in register D102   Torque in register D103
Speed in register D201   Speed in register D202   Speed in register D203
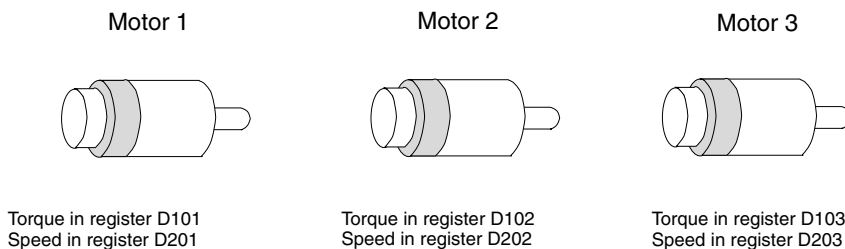
Register D0 is defined under **Setup/Index register** as Index register 1. The value in the register will determine for which motor the torque and speed are to be shown.
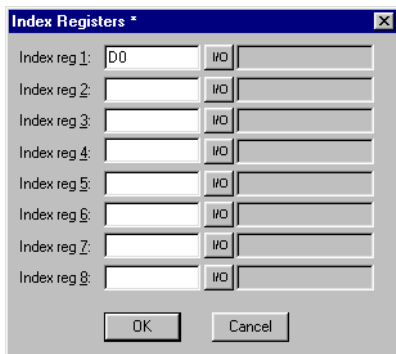
If the value is 1 in D0, the torque and speed for Motor 1 will be shown. If the value is 2 the values for Motor 2 are shown, and if the value is 3 the values for Motor 3. The value in register D0 is controlled with a Multiple choice object, where the texts Motor 1, Motor 2 and Motor 3 are shown. These 3 choices are also made maneuvrable.
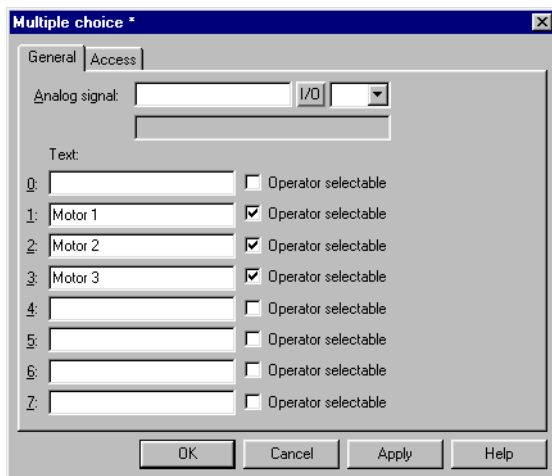
The torque and speed are shown in two numerical objects. In the object for the torque, D100 is specified as the analog signal and I1 as the index register.
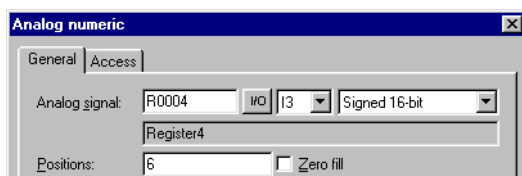
In the object showing the speed, D200 is specified as the analog signal and also here I1 as the index register.

With the maneuvrable Multiple choice object it will then be possible to select in run mode the alternatives Motor 1, Motor 2 and Motor 3. Depending on this choice, the values will be set to 1, 2 and 3 in register D0. The value in D0 is added to addresses for the objects which show the torque and speed. These can show the values in register D101, D102 or D103, and D201, D202 or D203.

NOTES

▮ Other suffixes can also be specified in addition to index register. The index register will not be counted twice if 32 bit registers are used.

▮ If the terminal is connected to a BDTP network, the same index register must be stated in the server and the client because the indexing is made in the server driver.

15   Communication

Not applicable for E50.

15.1   Communication with two controllers (Dual drivers)

It is possible to use two different drivers in the terminal, which means that the terminal can communicate with two different controllers simultaneously. The controllers can be connected in series to the serial ports on the terminal.
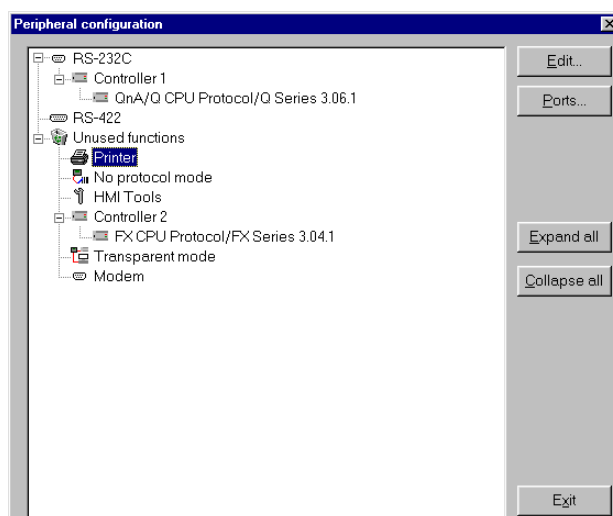
Addressing of the signals in the controllers is carried out as normal for each respective controller in accordance with the manual for the driver.

Select **File/Project Settings** in the programming tool.
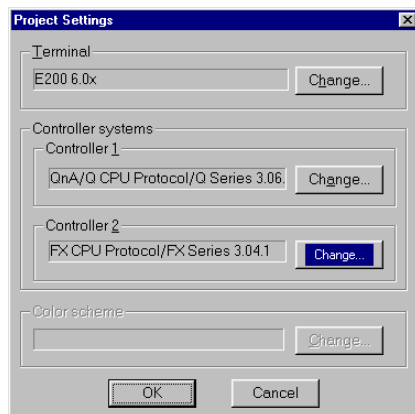
Select **Controller** by clicking on **Change** for **Controller 1** respective **Controller 2**. If the selected driver for **Controller 1** does not support dual drivers you cannot select a driver for **Controller 2**. Click on **OK**.

Select **Setup/Peripherals**. Drag **Controller 1** and **Controller 2** to the connection ports where the respective controllers are connected. It is possible to use RS-232C, RS-422. For further information on the connection of controllers to the terminal, refer to the manual for respective drivers.

15.1.1   Addressing

Addressing of the signals in the controllers is carried out as normal for each respective controller in accordance with the manual for the driver. The controller to which a created object is to be linked is selected by clicking the button corresponding to the required controller (1 or 2) in the tool field in the programming tool. Controller 1 is always pre-set when a project is created or opened.

If button 1 is pressed the signal is linked to an object created for controller 1 and if button 2 is pressed the signal is linked to an object created for controller 2.

It is also possible to click on the I/O button in the object created and select which controller the object is to be connected to from the I/O Browser.

To address a signal in controller 2 when controller 1 is preset, the signal must be given the addition @2 and correspondingly @1 for controller 1 if controller 2 is preset.

Example   Controller 1 is preset. Register D0 in controller 2 is to be connected to a potentiometer. D0@2 is entered in the field Analog signal in the dialog for the potentiometer.
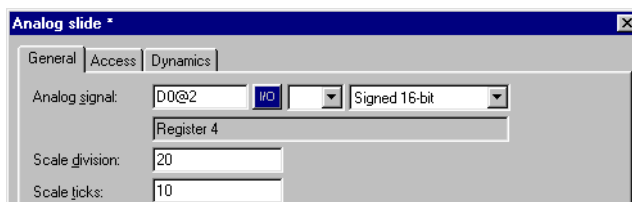
I/O Crossreference

The function I/O crossreference to show I/Os in a well-arranged manner can be used for both controller 1 and 2. The crossreference shows the preset controller.

Name list

The name list with attendant functions is supported for both controller 1 and 2.

NOTE   ▌ If communication with a controller is lost the communication with the functioning controller will be negatively affected, since the terminal will try to resume communication with the defective controller. In principle no communication with the terminal will function.
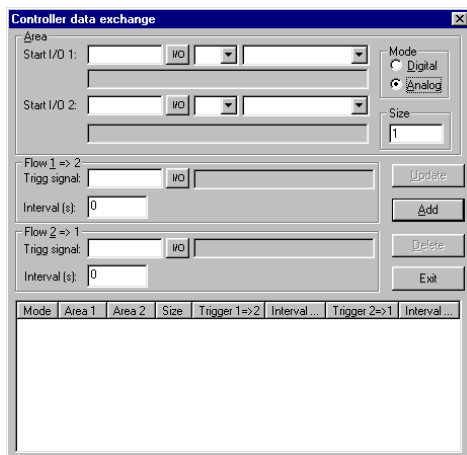
15.1.2   Examples of configurations where dual drivers are used in the terminal:

The terminal communicates with two controllers of different makes:
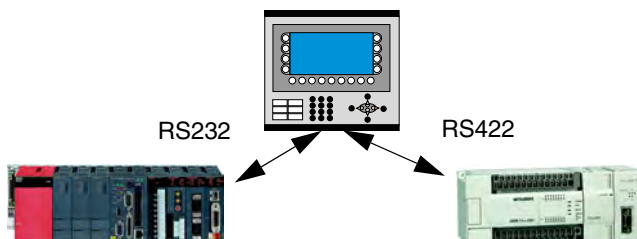
15.2   Data exchange between controllers

When two controller are connected to one and the same terminal (dual drivers are used in the terminal) it is possible to exchange data (analog and digital signals) between the controllers. The signals do not have to be of the same type in the two controllers. The exchange of data takes place in a virtual data channel, which goes between controller 1 and 2. You can define eight different data channels. The transfer of data can take place at intervals or during events. The conditions for how the transfer is made and which interval of signals are to be transferred for each data channel are specified under **Functions/Data exchange**.

Parameter	Description
Area	Start I/O 1 - Start address for the data channel for controller 1. (The subsequent fields are used to specify an index register and signal format.) See chapter *Index addressing* and *Signal format*.) Start I/O 2 - Start address for the data channel for controller 2. (The subsequent fields are used to specify an index register and signal format.) See chapter *Index addressing* and *Signal format*.)
Mode	Specify if the signals for the data channel are digital or analog.
Size	Specify the number of signals to be transferred (start address + subsequent) in the data channel. Max number of signals for a data channel is 255.
Flow 1 => 2	**Trigg signal** - A digital signal which controls the data transfer from controller 1 to controller 2 for the data channel. The signal status has the following implications: 0 - Inactive. 1 - Transfer – The terminal resets the signal when the transfer is completed **Interval** - Specifies the time in seconds between cyclical transfers of data channels. The interval parameter should be zero if it is not used. If the value is greater than 0 the parameter has a higher priority than the Trigger signal, and therefore a trigger signal cannot trigger transfers. The maximum number of seconds is 65,535.
Flow 2 => 1	**Trigg signal** - A digital signal which controls the data transfer from controller 2 to controller 1 for the data channel. The signal status has the following implications: 0 - Inactive. 1 - Transfer – The terminal resets the signal when the transfer is completed **Interval** - Specifies the time in seconds between cyclical transfers of data channels. The interval parameter should be zero if it is not used. If the value is greater than 0 the parameter has a higher priority than the Trigger signal, and therefore a trigger signal cannot trigger transfers. The maximum number of seconds is 65,535.

Click **Add** when the settings for a data channel have been defined.

NOTE   ▮ The function Data exchange functions like any other function in the terminal and therefore gives the same priority as other functions. Example: If the terminal is occupied (occupied with performing other functions) and a request for Data exchange is made, this will affect the transfer time and the exchange of data between the controllers will take longer than usual.
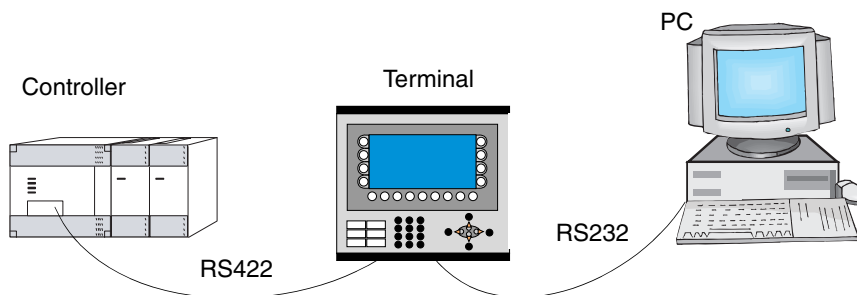
15.3 Transparent mode

Not applicable for E50.

In Transparent mode a communication port (programming/printer port) on the terminal which is not connected to the controller can be used to connect additional units working in parallel with the controller. These units can be terminals or a PC with programming tools for the controller or a master operator system.

For information on whether Transparent mode will function together with the connected controller, refer to the respective driver manual.

15.3.1 Connection of PC or other computer system

A PC with a programming tool or another computer system is connected directly to a communication port on the terminal, in this case the programming/printer port.

Terminal connected to the programming/printer port.

Settings in the terminal and PC

The following settings must be made in the terminal and PC for Transparent mode to function:

In the terminal project in the programming tool the communication settings are made under **Setup/Peripherals**. Drag the unit Transparent mode to the required communication port (to the port where the PC is to be connected to the terminal). Click with the right mouse button on the unit to set Transparent mode (if this is supported by the driver, see the manual for the driver). The settings for the port where the PC is connected must correspond with the settings in the program in the PC (the programming tool for the controller).

Parameter	Description
Mode	Select communication status Transparent or Passthrough. See section *Passthrough mode* for information on Passthrough mode. Timeout – Specify a time in seconds after which the terminal will return to RUN mode from the Passthrough mode if no Passthrough communication has occurred.

MITSUBISHI ELECTRIC

15.3.2   Connection of two terminals in Transparent mode

Additional terminals can be connected to the same controller with Transparent mode. We describe here how two terminals are connected. Additional terminals are connected in a similar way.

Cable connections

When two terminals are to be connected to one controller the first terminal is connected in accordance with the description in the Installation Manual. A cable is connected between the two terminals as shown in the following illustration. The cable is connected between the free port on the first terminal and the corresponding port on the second terminal. If the distance is more than 15 metres for RS232 communication, a signal amplifier must be used.

Cable between two terminals for RS232 communication

Cable between two terminals for RS-422 communication
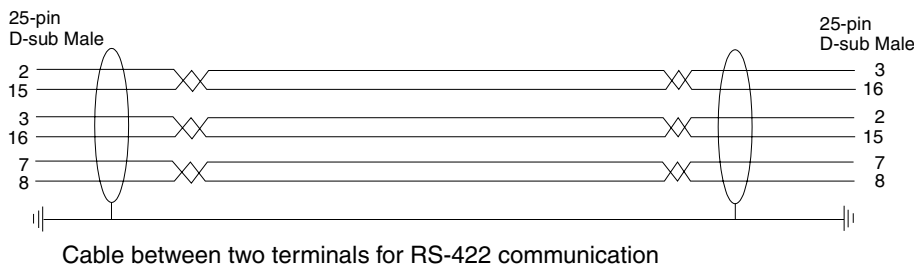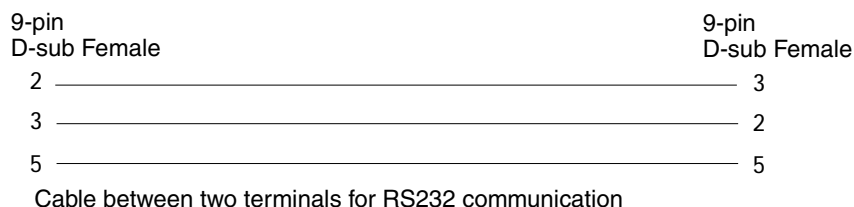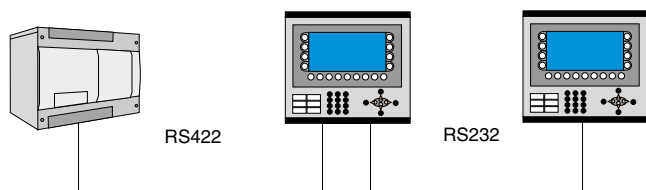
Settings in the first terminal

In the programming tool the communication settings are made under **Setup/Peripherals**. The settings for the port connected to the controller are set as normal. The settings for the port connected to the other terminal are optional.

Settings in the second terminal

In the programming tool the communication settings are made under **Setup/Peripherals**. The controller should be placed on the port used to connect the second terminal to the first terminal. The settings for this port should be the same as for the first terminal on the port where the second terminal is connected.

Transfer speeds

The transfer speed can vary between 600-57,600 baud. It is recommended to use the highest possible transfer speed between the terminals to achieve the best performance. The communication speed decreases with the number of connected terminals according to the following table.

Number of terminals	Access time to controller			
	Terminal 1	Terminal 2	Terminal 3	Terminal 4
1	100%	-	-	-
2	50%	50%	-	-
3	50%	25%	25%	-
4	50%	25%	12,5%	12,5%

15.3.3 Connection of three terminals in Transparent mode

A third terminal can be connected with a cable to Terminal 2 in the network as shown in the above diagram. The settings for the third terminal should be the same as the settings in the second terminal.
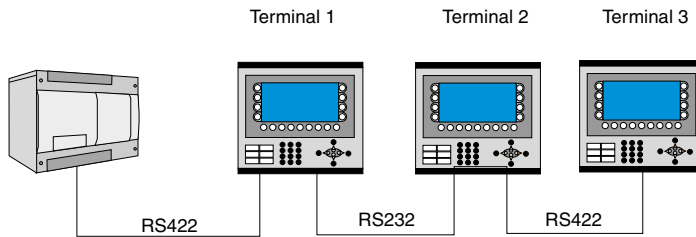
15.4   Passthrough mode

Not applicable for E50.

With the function Passthrough mode it becomes possible to set the terminal in a mode which enables communication between the programming tool for the connected controller on a PC and the controller to be established through the terminal. This is a joint function with the Transparent mode function and in similarity with Transparent mode only supports one controller. This means that Transparent or Passthrough mode can only be used on one of the terminal's communication ports. When Passthrough mode is activated and the PC communicates with the controller through the terminal, communication between the terminal and connected controller will stop. This is the difference between Passthrough and Transparent mode. A terminal where communication in Passthrough mode is in progress becomes locked for the operator, and only shows an empty display with the text Passtrough.
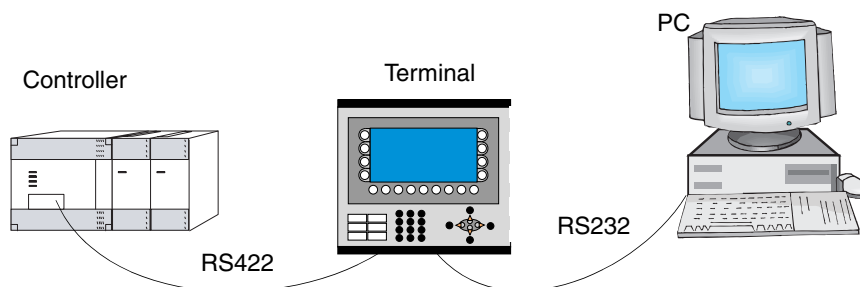
Terminal connected to the programming/printer port.

Passthrough mode for one of the connected controllers is activated and deactivated from the program HMI Tools Connect, which is available as an icon in the program group HMI Tools. See the manual for HMI Tools.

Settings in the terminal and PC

The following settings must be made in the terminal and PC for Passthrough mode to function:

In the terminal project in the programming tool the communication settings are made under **Setup/Peripherals**. Drag the unit Transparent mode to the required communication port (to the port from which the PC is to be connected to the terminal). Click on the unit with right mouse button to set Passthrough mode. The settings for the port where the PC is connected must correspond with the settings in the program in the PC (the programming tool for the controller).
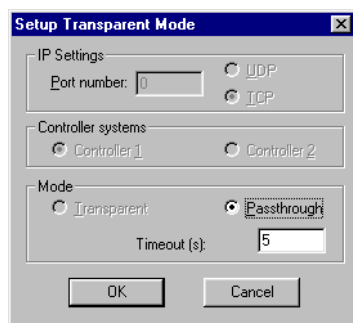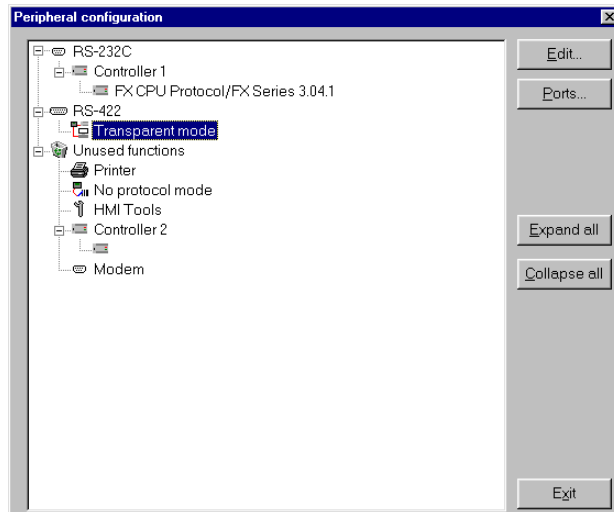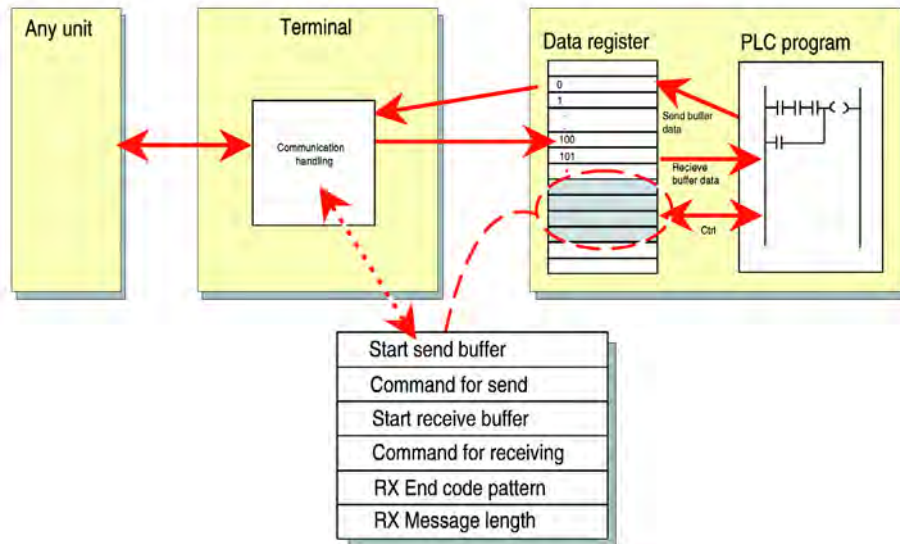

Parameter	Description
Mode	Select communication mode **Transparent** or **Passthrough**. See section Passthrough mode for information on Passthrough mode. **Timeout** – Specify a time in seconds after which the terminal will return to RUN mode from the Passthrough mode if no Passthrough communication has occurred.

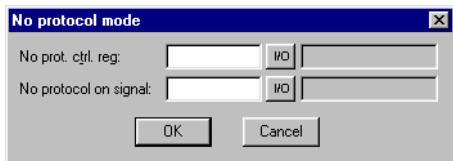Passthrough mode can be used both through serial and Ethernet communication.

15.5 The terminal as a communication interface (No protocol mode)

Not applicable for E50.

No protocol mode is used to connect different controller, or to connect external units such as barcode scanners or weighing instruments to the controller. The controller controls what is to be sent to the communication port. Data entering the communication port is entered directly in the controller register. Communication takes place with the transfer of the data register area according to the following control block.

MITSUBISHI ELECTRIC

Click on the right mouse button on **No Protocol Mode** under **Setup/Peripherals**.

| Register	Description
Ctrl reg 0	*Start register for the transmission data buffer*. The first register in the buffer area contains the total number of bytes to be sent. The following registers contain the data to be sent. The maximum buffer size is 127 registers = 254 bytes.
Ctrl reg 1	*Command register for transmission*. Set to 1 by the controller when transmission is requested. Set to 0 by the terminal when transmission is completed.
Ctrl reg 2	*Start register for the reception data buffer*. The first register in the bufferarea contains the total number of bytes received. The following registers contain the received data. The maximum buffer size is 127 registers = 254 bytes.
Ctrl reg 3	*Command register for reception*. - Set to 0 by the controller 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 controller when clearing of the port buffer is required. - Set to 3 by the terminal when clearing of the port buffer is completed. 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.
Ctrl reg 4	*End code* (1 or 2 bytes) on the received message.
Ctrl reg 5	*Length* of the received message. If 0, the end code is used.

Under **Setup/Peripherals** you drag the unit No protocol mode to the communication port you want to use. Click on the right mouse button to define which register is to be the first control register in the transfer area. This and the five following registers are used as control registers.

The controller can switch between No protocol mode and Transparent/printer mode during run mode. Specify a digital signal in the dialog for the parameter No protocol signal.
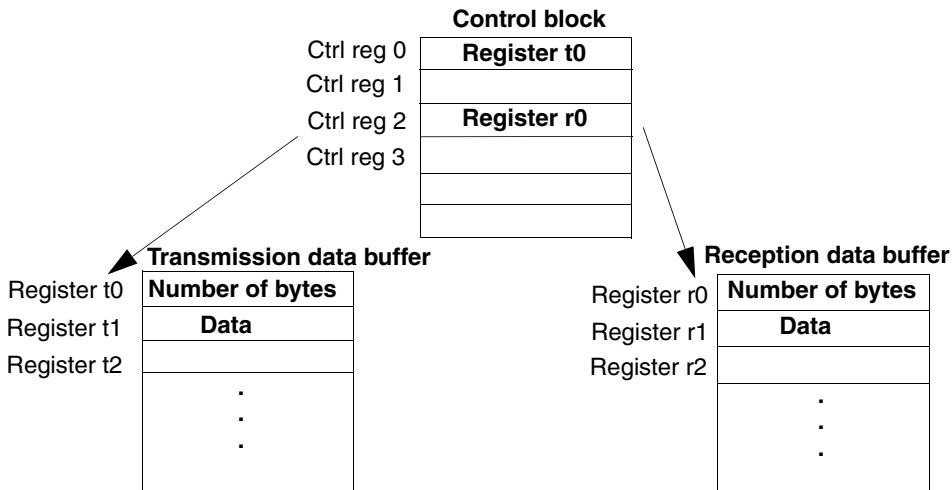
Example of using the No Protocol mode

The following example describes the use of the No Protocol mode for a weighing system.
The diagram below shows a three-stage block diagram for the communication.
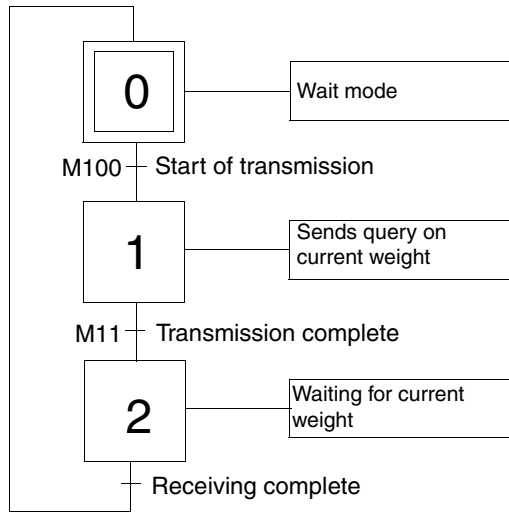
The terminal functions as a master system for the weighing system, i.e. it continuously asks for the current weight. The protocol appears as follows.

Terminal

│STX│?│CR│LF│

Weighing system

│STX│Weight in kg│CR│LF│

MITSUBISHI ELECTRIC

The protocol appears as follows.

Terminal

│STX│?│CR│LF│

Weighing system

│STX│Weight in kg│CR│LF│

The example is from the programming tool GX Developper for the Mitsubishi Electrics PLC system. D0 – D5 contains the commands regisers. D10 is the first register in the send buffer. D20 is the first register in the receive buffer.The program in the PLC system appears as follows:

```
    M8002
    ──┤├──────────────────────────────────┤ MOVP │ H3F02 │ D11 │──
     INIT                                               SEND
     PULSE                                              REG 1         ❶

    ──────────────────────────────────────┤ MOVP │ H0A0D │ D12 │──
                                                       SEND
                                                       REG 2
```
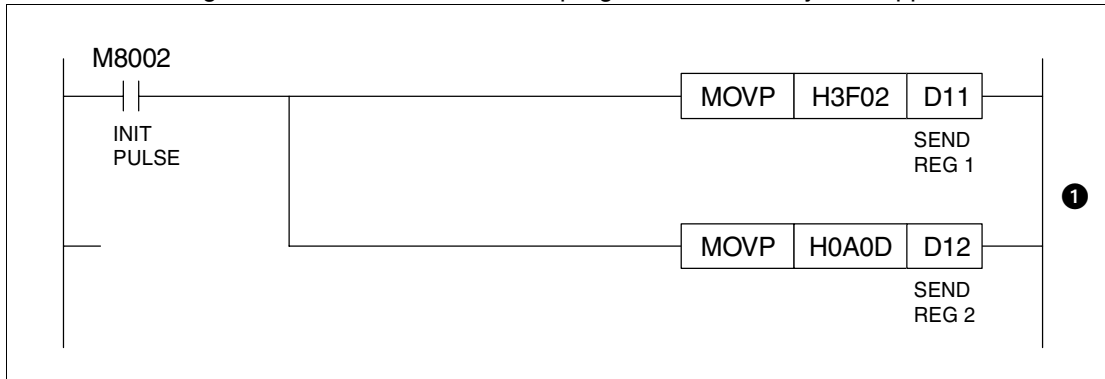
Number	Description
1	Initiation of question to the Weighing-machine: │STX│?│CR│LF│ Note the order of the commands in D11 and D12. D11=│?│STX│= H3F02 D12=│LF│CR│= H0A0D

Initiation of command registers D0 – D5

M8002				
11 —| |—	MOVP	K10	D0	❶
INIT PULSE			START REG SEND	
	MOVP	K20	D2	❷
		START REG RECE.	
	MOVP	HA0D	D4	❸
		END CODE	
	MOVP	K0	D5	❹
		RECE. LENGHT H	

Send/Receive-Sequence

M2	M14	M1		M0	❺
32 —| |—| |—|/|—		—( )—	
STEP2	RECE. OK	STEP1		STEP0	
M0					
—| |—				
STEP0					
M8002					
—| |—				
INIT PULSE					
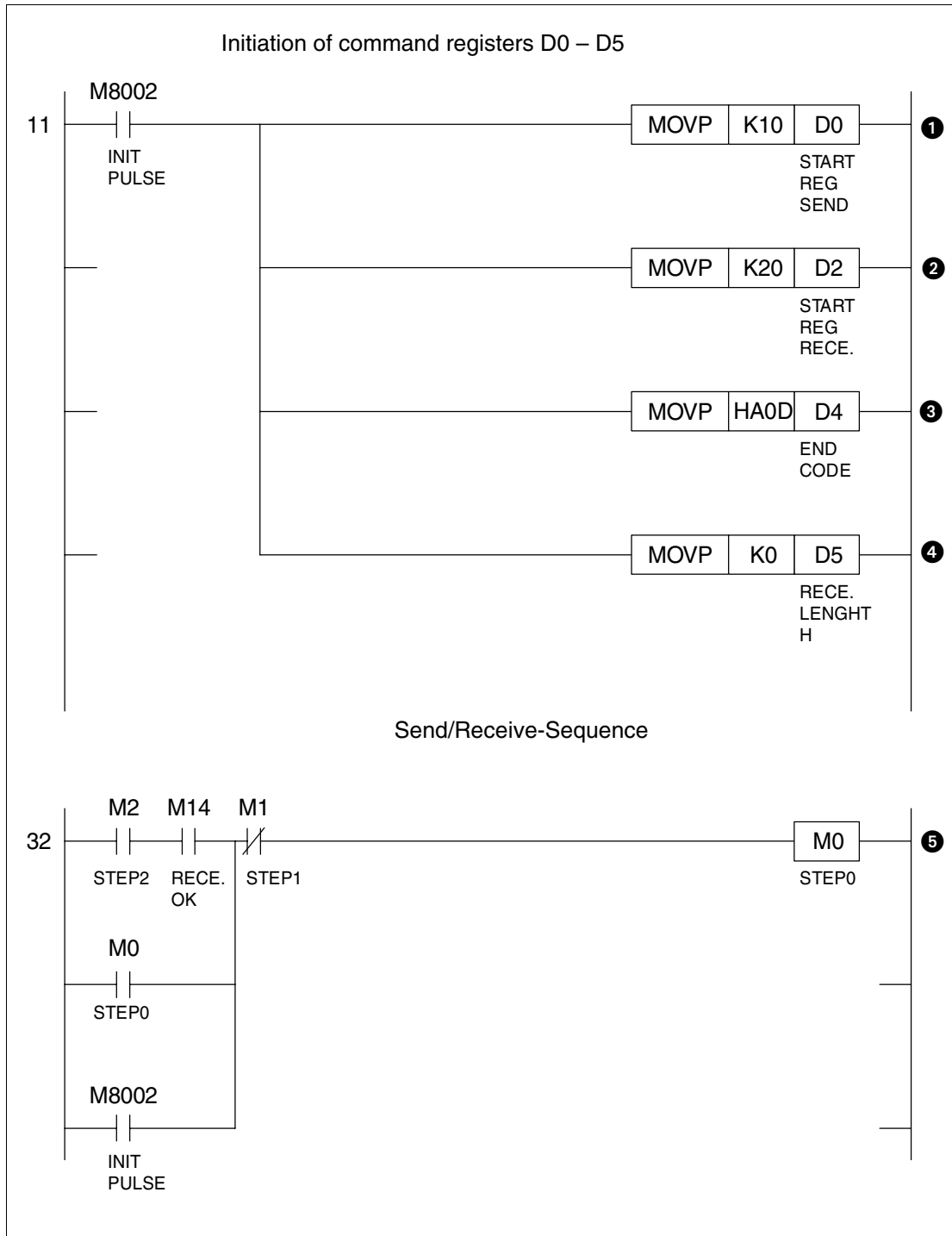
Number	Description
1	D0 = Start of the send buffer, K10 = D10
2	D2 = Start of the receive buffer, K20 = D20
3	D4 = End code /CR, CF
4	D5 = 0, use end code
5	Step 0 wait
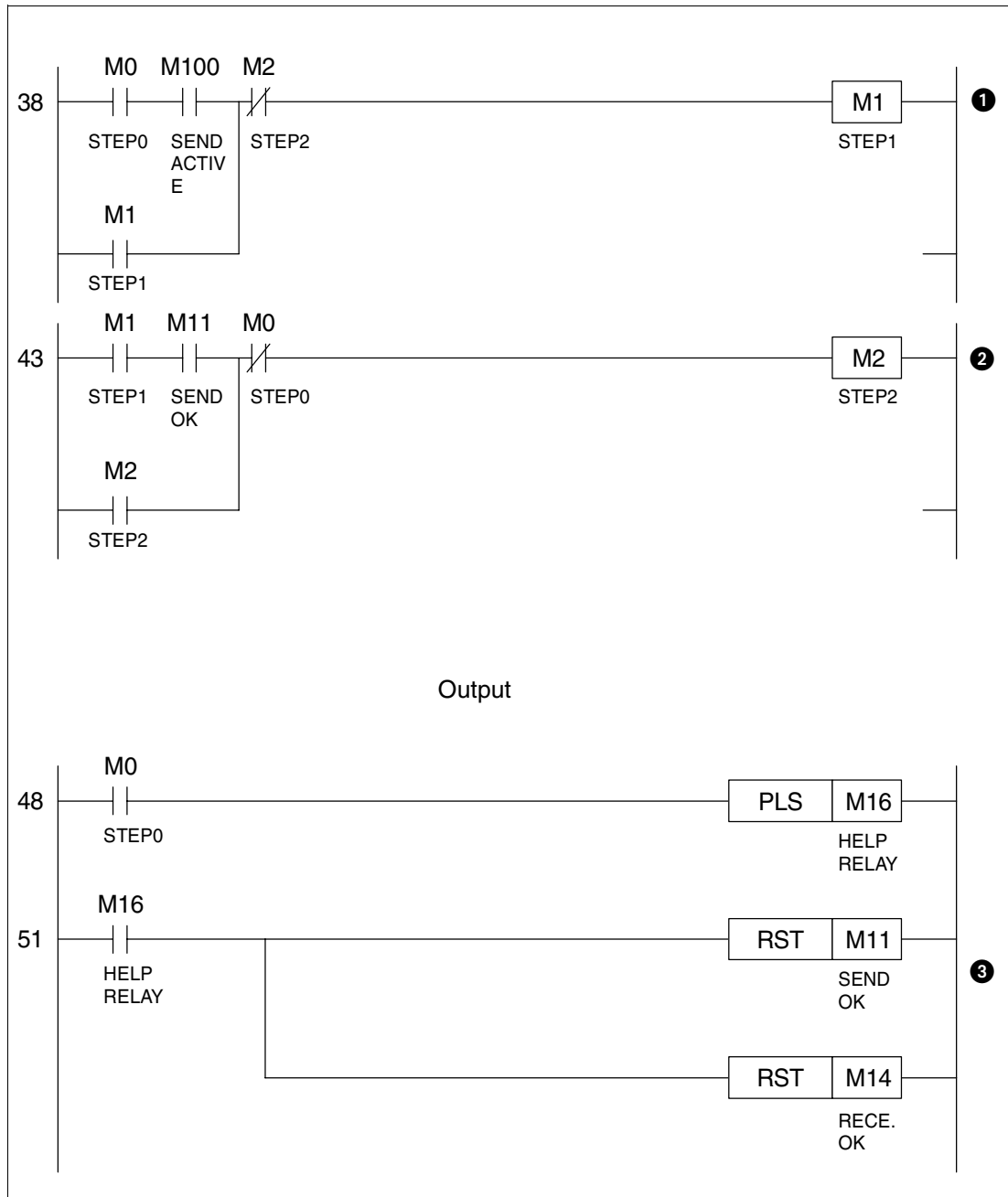
Number	Description
1	Step 1, send
2	Step 2, receive
3	Step 0, wait First program cycle in the wait state the send "ok" signal and the receive "ok" signal are cleared.

Step 1, transmission of question

| 54 | M1 ⊣⊢ STEP1 | | MOVP | K4 | D10 NBR SEND BYTES | ❶ |

| 60 | M1 ⊣⊢ STEP1 | | MOVP | K1 | D1 COMM. REG SEND | ❷ |

| 66 | M1 ⊣⊢ STEP1 | CMP | K0 | D1 COMM. REG SEND | M10 | ❸ |

| 74 | M1 ⊣⊢ STEP1 | | MOVP | K0 | D3 COMM. REG RECE. | ❹ |

Step 2, receiving weight

| 80 | M2 ⊣⊢ STEP2 | CMP | K1 | D3 COMM. REG RECE. | M13 | ❺ |

| 88 | | END |
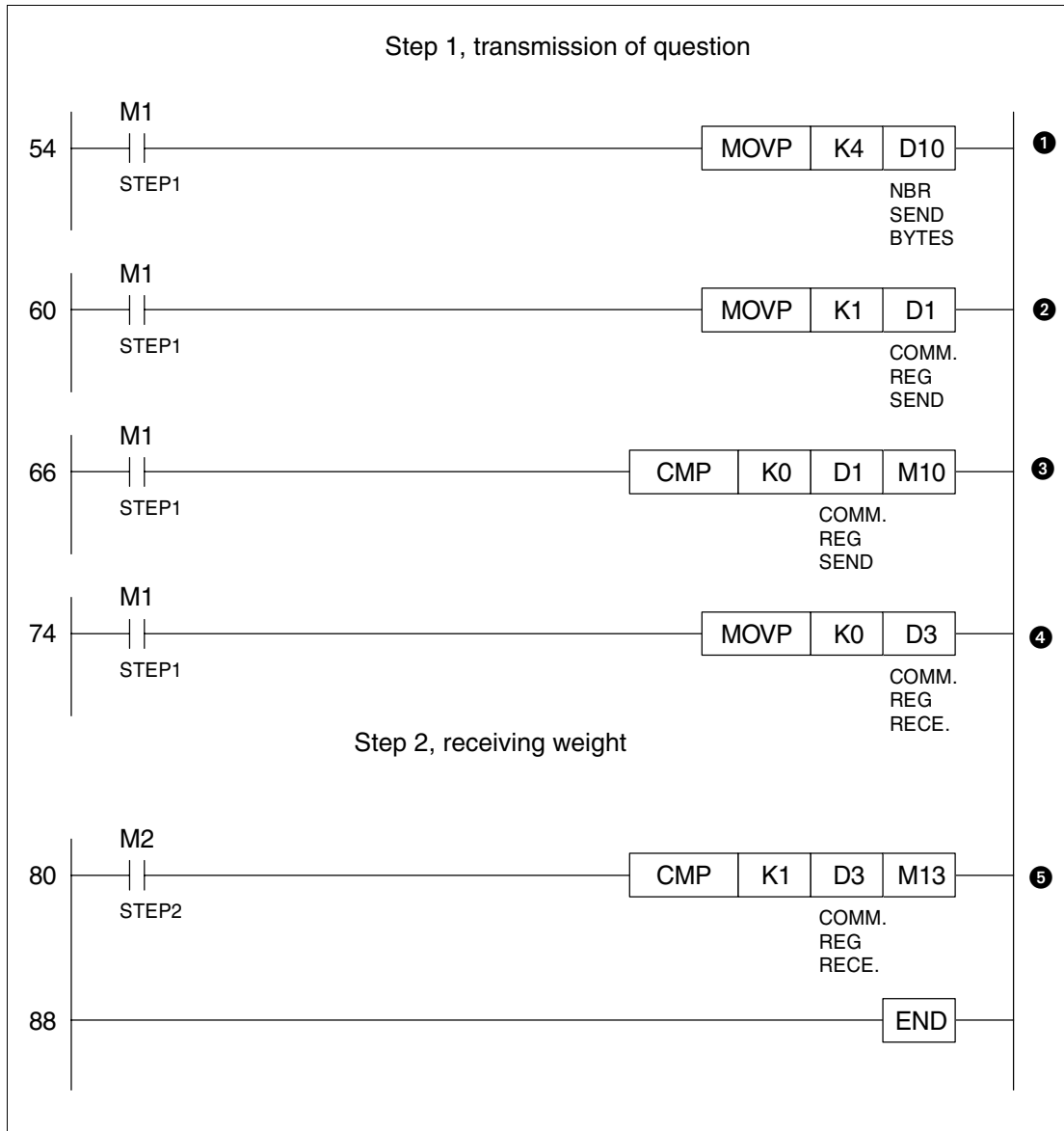
Number	Description
1	D10 = 4; number of bytes (signs) which will be sent to the weighing-machine
2	D1 = 1; activates transmission of question to the weighing-machine
3	Comparison if D1 = 0, if D1 = 0 the transmission is finisshed
4	Activate the receive buffer
5	Comparison if D3 = 1, if D3 = 1 a new weight is available. The weight in ASSCII-code is placed in D21 and forward.

MITSUBISHI ELECTRIC

15.6   Modem connection

Not applicable for E50.

A modem can be used to create a call-up connection to a PC. The settings for the connection are made under **Setup/Peripherals**. Show the dialog by marking **Modem** and clicking on **Edit...** or by clicking the right mouse button.
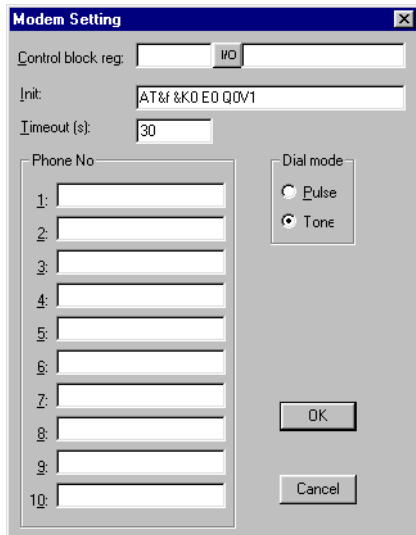
Control block reg:

The communication is established with three control registers in a control block. The first register in the control block is defined in the **Control block reg** in the dialog. The function of the control registers is described in the following table.

Register	Description
Ctrl reg 0	Contains commands describing how the controller is to call up and establish communication. 0 Wait for command. 1-10 Call up with the phone number entered in the field Phone No 1-10. Maximum of 40 characters. 11 Call up with a phone number stored in the controller. This number is stored as an ASCII string, which begins in the third control register and the following registers. The string can contain a maximum of 40 characters, i.e. 20 registers. It is not necessary to use all the registers. The last register read must contain the ASCII code 0. 101-110 An initiation string is sent to the modem. Enter Hayes modem command commands in the field Phone No 1-10. Command 101 sends the string in the field Phone No 1, etc. 111 An initiation string stored in the controller is sent to the modem. Enter Hayes modem commands beginning in the third control register. See command 11 for details. 255 Hang up command.

Register	Description
Ctrl reg 1	The second control register is used as a status register. It contains the results of the modem commands. The status register can contain the following: Status codes 0 Command executed successfully. 1 Dialling in progress. 2 Modem has established connection. 3 Modem detected a ring signal. Error codes 101 No connection. 102 Modem detected lost carrier. 103 Unspecified error from modem. 104 Modem gets no dial tone. 105 Busy when dialling. 106 No answer when dialling. 107 No reply from modem. 255 Unknown error/status.
Ctrl reg 2	The terminal can call up with a phone number stored in the controller. This number is stored as an ASCII string, which begins in the third control register and the following registers. The string can contain a maximum of 40 characters, i.e. 20 registers. It is not necessary to use all the registers. The last register read must contain the ASCII code 0.

Init:

Modem initiation string

Timeout

The number of seconds before the line is disconnected when idle. The default value is 30 seconds. Times between 5 and 600 seconds can be given.

Dial mode:

Select between Pulse and Tone.

Phone numbers 1-10:

Complete phone number for call up.

16 LEDs

Only applicable for E200.

This chapter is applicable for terminals with LEDs.

The terminal has built-in LED, which are connected to a register defined under **Functions/ LEDs**. The contents in the register determine the color, and appropriate blinking function of the LEDs according to the following table.

Register value (Hex)	Register value (Dez)	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

MITSUBISHI ELECTRIC

17 Function keys

This chapter describes how the function keys are used. A function key is linked to a signal in that its address is given after the respective key or by selecting a function from the list of options. The signal linked to a function key is activated according to the function indicated when the function key was defined.

NOTE ▮ It is only possible to activate two signals connected to function keys at the same time. I.e. if more than two function keys are pressed at the same time only the first two will be activated.
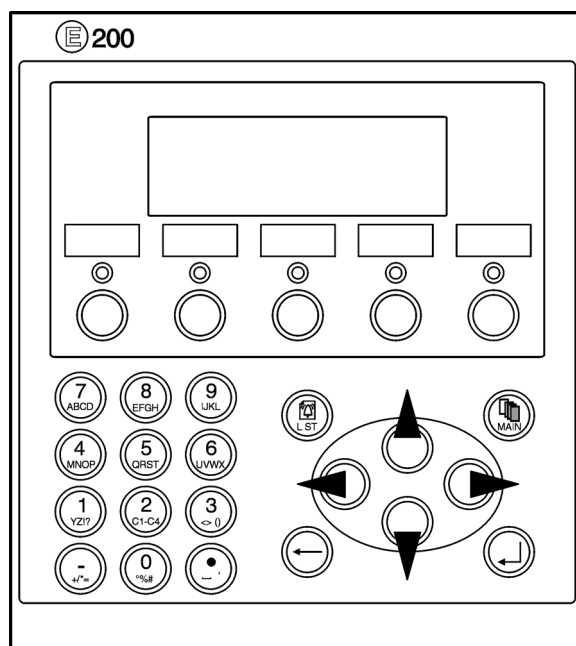
The terminals have a different number of built-in function keys, depending on the terminal model.

17.1   Definitions

There are two ways of defining function keys; *globally* and *locally*. Global function keys are defined and used in the complete application, i.e. they are valid in all blocks. Local function keys are defined and used in one block. A global definition is always accessible in run mode, on the assumption that the block shown in the display does not have any local definitions for the actual function key. Local definitions have a higher priority than global definitions. Global definitions are defined under **Functions/Function Keys**. Local function keys are defined in the relevant block's block header under F-keys.

Function	Description
I/O	The signal activated by the function key. (The subsequent field is used to specify an index register and signal format. See chapter *Index addressing* and *Signal format*.)
Event	With the function **Event** it is possible to select how the key will activate the given signal. Under Event there are the following alternatives: **Momentary** means that the signal is set to one as long as the key is active. **Toggle** means that the signal is set to one or reset alternately when the key is activated. **Set to one** means that the signal is set to one when the key is activated, and remains set to one. **Reset** means that the signal is reset when the key is activated, and remains reset. **Grouped** means that all signals belonging to a function key with the relevant group number are reset. The group number is given in the field **Group No**. A maximum of 8 function keys can be included in a group. **Set Analog** means that the analog signal linked to the function key is allocated the value given in the field **Value**. **Inc analog** means that the analog signal linked to the function key will increase the value by the value given in the field **Value**. **Dec analog** means that the analog signal linked to the function key will decrease the value by the value given in the field **Value**.
Set analog object to	Allocates maneuvrable analog object selected with the cursor the entered value.
Increment analog object with or set digital object to one	Increases the value for the selected maneuvrable analog object with the entered value or sets the selected maneuvrable digital object to one.

🔺 **MITSUBISHI ELECTRIC**

Function	Description
Decrement analog object with or resets digital object	Decreases the value for the selected maneuvrable analog object with the entered value or resets the selected maneuvrable digital object.
Set digital object momentarily	Sets the selected digital object to one as long as the key is pressed.
Jump to block	Jump to block with the given name/number.
Security level	The function keys can be defined with a security level. The security level means that the operator must login with a password for the same or higher security level to be able to use the function key.
Other function	The function key is linked to one of the functions in the list of options. See separate table "Other functions for function and touch keys".
Macro	The selected macro is run. See chapter *Macro*. With the button Edit macro you change the name of the selected macro or change the macro event for the selected event.

17.1.1 Other functions for function keys

Function	Description
Load recipe	Collect recipe from the memory in the terminal.
Save recipe	Save recipe to the memory in the terminal.
Delete recipe	Delete recipe from the memory in the terminal.
Append recipe	Adds signals and their values from the current block to an existing recipe. See chapter *Recipe handling*
Login to specified security level	Login. See chapter *Password*.
Logout	Logout.
Change login password	Change password.
Scroll one page up	Scroll the page in text block and in alarm list.
Scroll one page down	Scroll the page in text block and in alarm list.
Zoom up text size	Increase text size in the alarm list.
Zoom down text size	Reduce text size in the alarm list.
Acknowledge alarm	Acknowledge alarm in the alarm list.
Show alarm list	Show alarm list (block 990).
Jump to info block connected to the alarm	Jump to a block linked to the alarm. Applicable for selected alarm line or in the alarm list. See chapter *Alarm handling*.
List alarm groups	Select from which alarm group the alarm should be shown in the alarm list.
Return to previous block	Show "previous block" functions in nine levels backwards. When block 0 is shown it is not possible to jump to the previous block with this function. Block jumping cannot be made with this function if log on is made in run mode at a higher security level than the current level.
Jump to main block (block 0)	Shows the start block, block number 0.
Show object info	Shows min and max values for analog object in the Text block in operating mode.
Enter	Corresponds to pressing the Enter key.
Show diagnostic page	Shows the diagnostics page. See chapter *Terminal functions*.
Change recipe directory	Edit recipe directory in the terminal. See chapter *Recipe handling*.
Make recipe directory	Create recipe directory in the terminal. See chapter *Recipe handling*.
Delete recipe directory	Delete recipe directory in the terminal. See chapter *Recipe handling*.

17.2 Jump to block with function keys

This function makes it possible to use function keys to jump to blocks without having to use **Display signal**. When the key is defined, locally or globally, you select **Jump to block** in the select list.

Changing blocks through function keys is the quickest way of changing blocks, and does not occupy digital signals in the controller.

MITSUBISHI ELECTRIC

18   Macro

Not applicable for E50.

This chapter describes how you create macros, i.e. link several events in the terminal together to one single command. If you often perform the same commands and settings in the terminal you can use macros to automate these maneuvres, i.e create a macro for them. A macro is activated from local and/or global function keys. The function Macro is activated under **Functions/ Macro**.
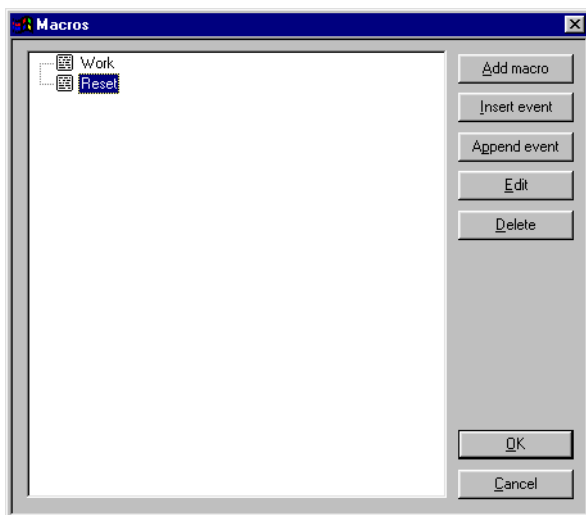
18.1   Add macro

When you push the button **Add macro,** the following dialog is shown. There is no limitation to the number of macros that can be defined.
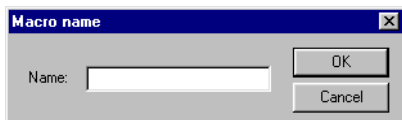
Specify an optional name for the macro, the name must be unique. When you click **OK** the macro is shown with the specified name in the list.

18.1.1   Insert event/Append event

When you push the button **Insert event/Append event,** the following dialog is shown. Each macro can contain max eight different events (lines).

Parameter	Description
I/O	Specify the signal to be linked to an event in the macro. In the field Event, select which event is to be linked to the signal in the macro. You can choose between the following events: **Set** - means that the digital signal is set to one and remains set to one when the macro is activated from a key. **Grouped** means that signals belonging to a function key with the current group number are reset. The group number is given in the field, **Group No**. A maximum of 8 function keys can be included in a group. **Dec analog** - means that the analog signal will decrease the value by the value indicated in the field **Value** when the macro is activated from a key. **Reset** - means that the digital signal is reset and remains reset when the macro is activated from a key. **Set Analog** - means that the analog signal is given the value by the value indicated in the field **Value** when the macro is activated from a key. **Toggle** - means that the digital signal is set to one and reset alternately when the macro is activated from a key. **Inc analog** - means that the analog signal will increase the value by the value indicated in the field **Value** when the macro is activated from a key.
Jump to block	Indicate the number of the name of the block to which a block jump is to be made when the macro is activated from a key. A block jump can only be the last event in a macro, since a block jump ends the macro.

18.1.2   Edit

With the button **Edit** you change the name of the selected macro or change the macro event for the selected event. You can also double click on the macro or macro event to edit.

18.1.3   Activate macro

The macro is activated from the function keys. Each key (global or local) can be linked to a macro. The required macro for the key is selected in the dialogs for local and global function keys.
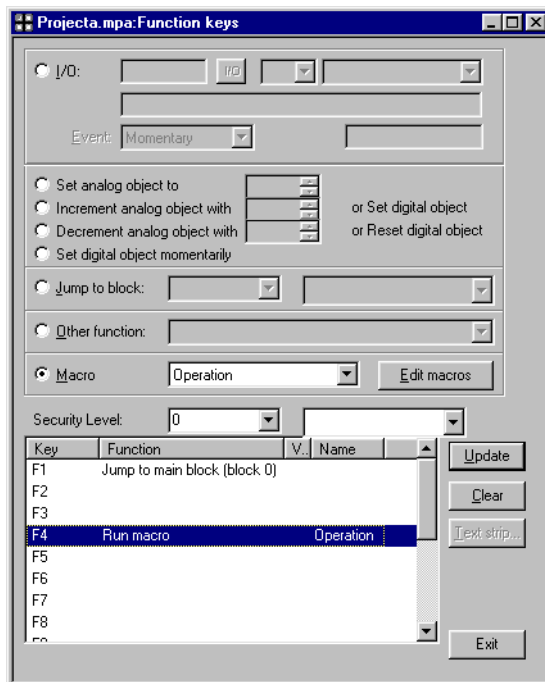
19   Transferring projects

To use a project in the terminal it has to be transferred from the PC where it is programmed to the terminal.

Connect the PC where the programming tool is installed to the terminal with cable CAB5 (MAC PC CAB R2) or CAB6.

CAB5 (MAC PC CAB R2)
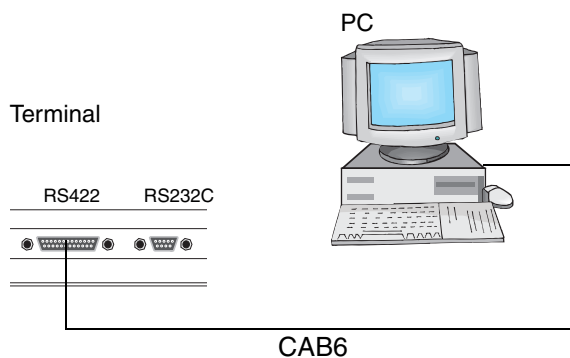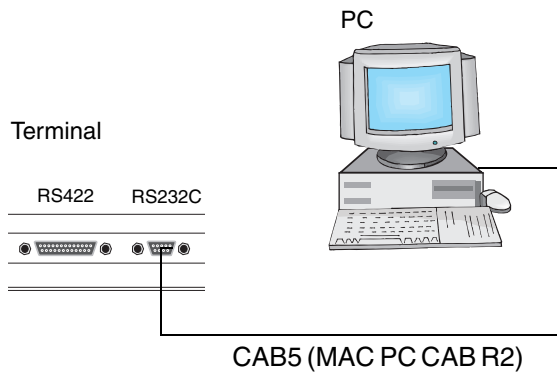
CAB6

NOTES   ▮ The cable CAB6 cannot be used to transfer projects between E50 and a PC.

▮ When a project is transferred to E50 the driver and system program are also transferred. All communication between the PC and E50 is conducted through E-Designer.

19.1   Installation of the terminal

It is not normally necessary to install the terminal. The transfer of projects is controlled from the programming tool. When necessary the transfer parameters are set in the terminal in configuration mode under **Setup/Port Parameters/E-Designer**.

NOTE   ▮ The communication settings must be the same in the programming tool as in the terminal.

19.2   Transfer settings

The transfer of projects is controlled from the programming tool. In the programming tool under **Transfer/Project** you can select what is to be transferred.
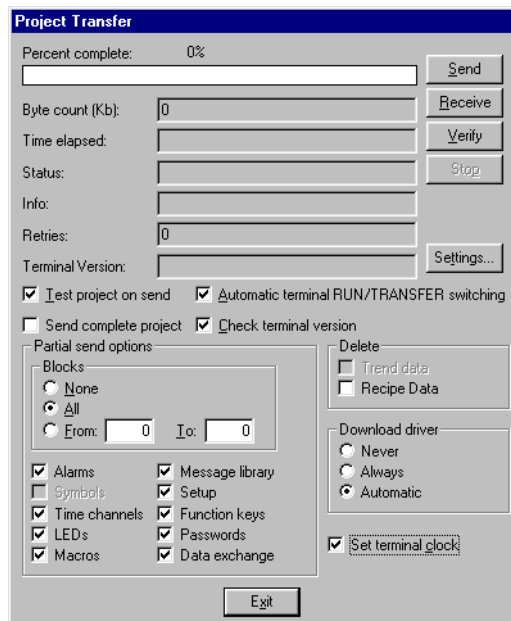
Parameter	Description
Percent complete	Shows how much of the project has been transferred during the transfer.
Byte count (Kb)	Shows how many kilobytes have been transferred during the transfer.
Time elapsed	Shows how much time has elapsed since the functions Send, Receive or Verify were activated.
Status	Shows the transfer status and which part of the project is being transferred. E.g. settings, block, alarm groups, symbols and function keys.
Info	Shows the defined driver to be transferred to the terminal.
Retries	In the event of transfer problems the programming tool makes a number of retries before the transfer shuts down.
Terminal Version	When contact has been established with the terminal, the current terminal type and its system program version number are shown.
Test project on send	Automatically tests the project before transfer.
Automatic terminal RUN/TRANSFER switching	The terminal is automatically set in transfer mode and returns after transfer to the mode the terminal was in before the transfer.
Check terminal version	Compares the system program version in the terminal with the installed version for the project in the programming tool.
Send complete project	Select whether the complete project is to be transferred.

Parameter	Description
Partial send options	Block None:No blocks to be sent to terminal. All:All blocks to be sent to terminal. From To:Specify an interval of blocks to be sent to the terminal. Alarms:Select if alarms are to be sent to the terminal. Symbols:Select if symbols are to be sent to the terminal. Time channels:Select if time channels are to be sent to the terminal. LEDs:Select if LEDs are to be sent to the terminal. Macros:Select if macros are to be sent to the terminal. Message library:Select if message library is to be sent to the terminal. Setup:Select if configurations under Settings are to be sent to the terminal. Function keys:Select if function keys are to be sent to the terminal. Passwords:Select if passwords are to be sent to the terminal. Data exchangeSelect if data transfer is to be sent to the terminal.
Delete	Recipe data:All stored recipe data in the terminal is deleted.
Download driver	Never:Driver never downloaded. Always:Driver always downloaded. Automatic:Driver will be downloaded to the terminal when the driver in terminal and the defined driver in the current project are not the same, or the same version. For E50 the parameters are applicable both for the driver and system program.
Set terminal clock	The clock in the PC is transferred to the terminal.
Send	Transfer the project to the terminal with the current settings.
Receive	The programming tool imports the project from the terminal. The current project in the programming tool is overwritten. There must be an active project in the programming tool if you are to upload a project from the terminal.
Verify	Checks that the project in the programming tool is the same as in the terminal.
Stop	Interrupts the transfer in progress.
Settings	Installation of transfer parameters. Must be the same as in the terminal. See table below.

The communication parameters are set under **Transfer/Comm Settings** or by selecting **Setup** in the dialog **Project Transfer**.

Settings in the dialog **Communication parameters**.

Parameter	Description
Use serial transfer	Select if serial transfer of the project is to be used. See section *"Serial transfer"*.
Use modem transfer	Select if modem transfer of the project is to be used. See section *"Modem transfer"*.
Port	Select communication port for the PC.
Baudrate	Transfer speed (bits/second).
Timeout (ms)	Indicate the number of milliseconds between retries.
Retries	Indicate the number of retries before the transfer shuts down.
Speed set manually	Only used for older terminal versions during modem communication. The transfer speed must then be set manually in the terminal and in the programming tool, they must be the same. The terminal must also be manually set in transfer mode.
Parity	Select type of parity control.
Databits	Number of databits for the transfer. Must be 8 bits.
Stopbits	Select the number of stopbits for the transfer.

NOTES   ▮ If other programs are run under Windows at the same time as a project is transferred, this can lead to communication errors. This is avoided by closing other programs.

▮ Links to signals will be included when transferring blocks.

19.3   Serial transfer

For serial transfer you select **Use serial transfer** under **Transfer/Comm Settings**. When you press **Send** in the dialog **Project Transfer** the project will be transferred to the terminal.

19.4   Modem transfer

Not applicable for E50.

To transfer by modem you select **Use modem transfer** under **Transfer/Comm Settings**. When you press **Send** in the dialog **Project Transfer** the project will be transferred to the terminal.

19.4.1   Modem settings

The settings on the modem connected to the operator terminal should be set as follows:

AT &F E0 Q1 &D0 &K0 &W

The settings on the modem connected to the PC should be set as follows:

AT &F &D0 &K0 &W

The table below describes the modem commands.

Befehl	Description
AT	Informs the modem that a command is coming. All command strings are preceded by AT.
&F	Loads the modem's default settings.
E0	Echo off.
Q1	Command answer off.
&D0	Modem ignores the DTR signal.
&K0	No flow control.
&W	Saves settings.

NOTE   I The modem must be set to **autoanswer** for the transfer to function.

19.4.2   Communication settings

① Configure the modem.

② Set the communication settings under **Transfer/Comm Settings** in E-Designer. Select Use modem transfer.
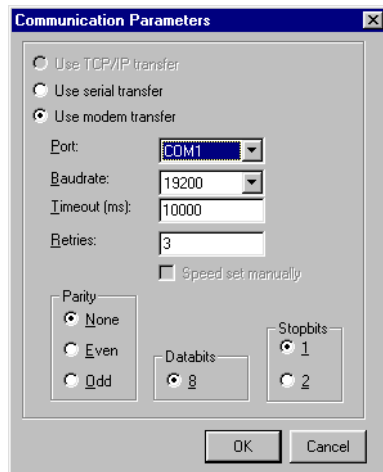
③ Select port, and set the transfer speed (baudrate), parity and stopbits.

④ Use the program HMI Tools Connect to call up.

⑤ Thereafter you select Transfer in E-Designer.
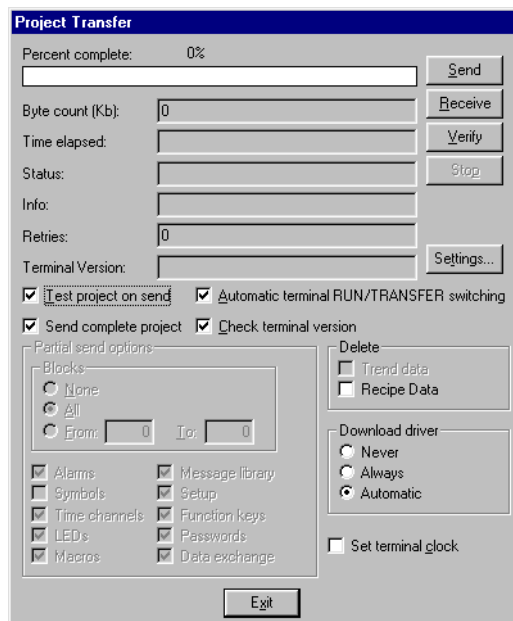
Select **Automatic terminal RUN/TRANSFER** switching.

NOTE   ❙ For information on modem communication and settings for modem communication with older terminal versions (system program), refer to the corresponding manual for the terminal version.

⟁ **MITSUBISHI ELECTRIC**

20   Technical data

E50

Parameter	Data
Front panel	Width 104 x height 69 x depth 4.5 mm.
Mounting depth	38 mm excl. D-sub and 38 mm incl. D-sub.
Panel cut out	S-02857
Front panel seal	IP 65, NEMA 4, NEMA 4X (indoor use only)
Rear panel seal	IP 20
Keyboard material	Membrane keyboard with polyester snap discs. Overlay film of Autotex F207 with print on reverse side. 1 million operations.
Reverse side material	Alu-Zinc
Weight, excl. D-sub	0.2 kg.
Serial port RS-422	25-pin D-sub contact, chassis-mounted female connector with standard locking screws 4-40 UNC.
Serial port RS-232C	9-pin D-sub contact, chassis-mounted male connector with standard locking screws 4-40 UNC, available when the attendant adapter 232-25-ADP is mounted on the RS-422 port.
Memory	Flash memory, 16 kb for application.
Power consumption at rated voltage	Max: 200 mA
Display	STN-LCD display, 2 lines of 16 characters, 5 mm character height. LED backlight lifetime 50,000 h at ambient temperature of +25°C. LED.
Active area of display, WxH	55,7 x 11,0 mm
Contrast setting	Through potentiometer on the top right corner on the back of the terminal.
Supply voltage	+5 VDC, ±5%, from controller or externally. Battery eliminator (connector 2.1 mm, centre pin 0V).
Ambient temperature	+0 ° to +50 °C
Storage temperature	-20 ° to +70 °C
Relative humidity	Max 85% non-condensing.
EMC tests on the terminal	The terminal complies with the requirements in paragraph four in the directive 89/336/EEC. Noise tested according to: EN50081-1 emission and EN61000-6-2 immunity.
UL approval	UL 508, UL 1604 (Class I Div 2, Groups A, B, C, D, T4)
DNV approval	Det Norske Veritas type approval certificate. Location classes: Temperature A, Humidity B, Vibration A, Enclosure C (panel front only).
MTBF (Mean time between failure) (h)	402 225

E100

Parameter	Data
Front panel	Width 142 x height 90 x depth 3.5 mm.
Mounting depth	29 mm excl. D-sub and 97 mm incl. D-sub.
Panel cut out	S4-03205
Front panel seal	IP 65, NEMA 4, NEMA 4X (indoor use only)
Rear panel seal	IP 20
Keyboard material	Membrane keyboard with polyester snap discs. Overlay film of Autotex F207 with print on reverse side. 1 million operations.
Reverse side material	Alu-Zinc
Weight, excl. D-sub	Excl. D-sub: 0.5 kg.
Serial port RS-422	25-pin D-sub contact, chassis-mounted female connector with standard locking screws 4-40 UNC.
Serial port RS-232C	9-pin D-sub contact, chassis-mounted male connector with standard locking screws 4-40 UNC.
Memory	Flash memory, 64 kb for application.
Real-time clock	±10 PPM + error resulting from ambient temperature and supply voltage. Total max error: 1 min./month = 12 min./year. Battery life for real-time clock: 10 years.
Power consumption at rated voltage	Max: 200 mA
Display	STN-LCD display, 2 lines of 20 characters, 5 mm character height. LED backlight lifetime 50,000 h at ambient temperature of +25°C..
Active area of display, WxH	73,5 x 11,5 mm
Contrast setting	Through potentiometer on reverse side of the terminal.
Supply voltage	+5 VDC, ±5%, from controller or externally. Battery eliminator (connector 2.1 mm, centre pin 0V).
Ambient temperature	+0 ° to +50 °C
Storage temperature	-20 ° to +70 °C
Relative humidity	Max 85% non-condensing.
EMC tests on the terminal	The terminal complies with the requirements in paragraph four in the directive 89/336/EEC. Noise tested according to: EN50081-1 emission and EN61000-6-2 immunity.
UL approval	UL 508, UL 1604 (Class I Div 2, Groups A, B, C, D, T4)
DNV approval	Det Norske Veritas type approval certificate. Location classes: Temperature A, Humidity B, Vibration A, Enclosure C (panel front only).
MTBF (Mean time between failure) (h)	345 155

MITSUBISHI ELECTRIC

E150

Parameter	Data
Front panel	Width 142 x height 100 x depth 3.5 mm.
Mounting depth	29 mm excl. D-sub and 96.5 mm incl. D-sub.
Panel cut out	S-03258
Front panel seal	IP 65, NEMA 4, NEMA 4X (indoor use only)
Rear panel seal	IP 20
Keyboard material	Membrane keyboard with polyester snap discs. Overlay film of Autotex F207 with print on reverse side. 1 million operations.
Reverse side material	Alu-Zinc
Weight, excl. D-sub	Excl. D-sub: 0.5 kg.
Serial port RS-422	25-pin D-sub contact, chassis-mounted female connector with standard locking screws 4-40 UNC.
Serial port RS-232C	9-pin D-sub contact, chassis-mounted male connector with standard locking screws 4-40 UNC.
Memory	Flash memory, 64 kb for application.
Real-time clock	±10 PPM + error resulting from ambient temperature and supply voltage. Total max error: 1 min./month = 12 min./year. Battery life for real-time clock: 10 years.
Power consumption at rated voltage	Max: 200 mA
Display	STN-LCD display, 2 lines of 20 characters, 5 mm character height. LED backlight lifetime 50,000 h at ambient temperature of +25°C.
Active area of display, WxH	73,5 x 11,5 mm
Contrast setting	Through potentiometer placed in the top right corner on the back of the terminal.
Supply voltage	+5 VDC, ±5%, from controller or externally. Battery eliminator (connector 2.1 mm, centre pin 0V).
Ambient temperature	+0 ° to +50 °C
Storage temperature	-20 ° to +70 °C
Relative humidity	Max 85% non-condensing.
EMC tests on the terminal	The terminal complies with the requirements in paragraph four in the directive 89/336/EEC. Noise tested according to: EN50081-1 emission and EN61000-6-2 immunity.
UL approval	UL 508, UL 1604 (Class I Div 2, Groups A, B, C, D, T4, Groups A, B, C, D, T4)
DNV approval	Det Norske Veritas type approval certificate. Location classes: Temperature A, Humidity B, Vibration A, Enclosure C (panel front only).
MTBF (Mean time between failure) (h)	345 155

E200

Parameter	Data
Front panel	Width 147 x height 163.5 x depth 5 mm.
Mounting depth	38 mm excl. D-sub and 107 mm incl. D-sub.
Panel cut out	S4-02809
Front panel seal	IP 65, NEMA 4, NEMA 4X (indoor use only)
Rear panel seal	IP 20
Keyboard material	Membrane keyboard with polyester snap discs. Overlay film of Autotex F207 with print on reverse side. 1 million operations.
Reverse side material	Alu-Zinc
Weight, excl. D-sub	Excl. D-sub: 0.7 kg.
Serial port RS-422	25-pin D-sub contact, chassis-mounted female connector with standard locking screws 4-40 UNC.
Serial port RS-232C	9-pin D-sub contact, chassis-mounted male connector with standard locking screws 4-40 UNC.
Memory	Flash memory, 64 kb for application.
Real-time clock	±10 PPM + error resulting from ambient temperature and supply voltage. Total max error: 1 min./month = 12 min./year. Battery life for real-time clock: 10 years.
Power consumption at rated voltage	Max: 150 mA
Display	STN-LCD display, 4 lines of 20 characters, 5 mm character height. LED backlight lifetime 50,000 h at ambient temperature of +25°C.
Active area of display, WxH	70.4 x 20.8 mm
Contrast setting	Through potentiometer placed in the top right corner on the back of the terminal.
Supply voltage	+24 VDC (20-30 VDC), 3-pin jack connection block. CE: The power supply must conform with the requirements for SELV according to IEC 950 or IEC 742. UL: must conform with requirements for class 2 supply voltages.
Fuse	1 AT, Littelfuse R452 001 (Nano^2 SMF Slo-Blo) or 800 mAT, 5 x 20 mm.
Ambient temperature	+0 ° to +50 °C
Storage temperature	-20 ° to +70 °C
Relative humidity	Max 85% non-condensing.
EMC tests on the terminal	The terminal complies with the requirements in paragraph four in the directive 89/336/EEC. Noise tested according to: EN50081-1 emission and EN61000-6-2 immunity.
UL approval	UL 508, UL 1604 (Class I Div 2, Groups A, B, C, D, T4)
DNV approval	Det Norske Veritas type approval certificate. Location classes: Temperature A, Humidity B, Vibration A, Enclosure C (panel front only).
MTBF (Mean time between failure) (h)	128 062

▲ MITSUBISHI ELECTRIC

20.1   Membrane keyboard

20.1.1   Solvent resistance for Autotex 2

The front material on the Autotex F207 operator terminal is resistant to the following substances for more than 24 hours exposure according to DIN 42 115 Del 2 without visible change..

Ethanol	Formaldehyde 37%-42%	1.1.1. Trichloroethane (Genklene)
Cyclohexanol	Acetaldehyde	Ethylacetate
Diacetone alcohol	Aliphatic hydrocarbons	Diethyl ether
Glykol	Toluene	N-Butyl acetate
Isopropanol	Xylene	Amylacetate
Glycerine	White spirit	Butycellosolve
Methanol	Fromic acic <50%	Ether
Triacetin	Acetic acid <50%	Sodium hypchlorite<20% (Bleach)
Dowanol DRM/PM	Phosphoric acid <30%	Hydrogen peroxide <25%
Acetone	Hydrochloric acid <36%	Potassium carbonate
Methyl ethyl ketone	Nitric acid <10%	Washing powder
Dioxan	Trichloracetic acid <50%	Rinsing agent
Cyclohexanone	Sulphuric acid <10%	Ferric chloride
MIBK	Cutting oil	Ferrous chloride
Isophorone	Diesel oil	Dibutyl phthalate
Ammonia<40%	Linseed oil	Diethyl phthalate
Caustic soda <40%	Paraffin oil	Sodium carbonate
Potassium hydroxide <30%	Blown castor oil	
Alkalicarbonate	Silicone oil	
Bicarbonate	Turpentine substitute	
Potassium ferrocyanide/ferricyanide	Universal brake fluid	
Acetonitrile	Decon	
Sodium bisulphate	Aviation fuel	
	Petrol	
	Teepol	
	Fresh water	
	Saltwater	

Autotex is resistant to glacial acetic acid for less than 1 hours exposure in accordance with DIN 42 115 Part 2 without visible change.

Autotex is not resistant to the following substances:

Concentrated mineral acids
Concentrated caustic solution
High pressure steam over 100°C
Benzyl alcohol
Methylene chloride

Autotex is resistant to the following reagents during 24 hours exposure at 50°C without visible discoloration:

Top Job	Grape Juice	Ariel	Ajax
Jet Dry	Milk	Persil	Vim
Gumption	Coffee	Wisk	Domestos
Fantastic		Lenor	Vortex
Formula 409		Downey	Windex

Very slight discoloration was noted during critical examination with the following substances:

Tomato juice
Tomato ketchup
Lemon juice
Mustard

20.1.2   Environmental data
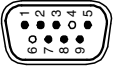
Outdoor use

In common with all polyester-based film, Autotex should not be exposed to direct sunlight for prolonged periods.

20.2   Drawings

20.2.1   E-series **RS-232/RS422**

RS-422

Pin no	Name	Signal direction Terminal<-> XXX
2	+TxD	↑
15	-TxD	
3	+RxD	↓
16	-RxD	
4	+RTS	↑
17	-RTS	
5	+CTS	↓
18	-CTS	
20	[1]	
21	[1]	
7, 8	0 V	
14	+5 V <50 mA	↑
12,13 24, 25	[2] +5 V <200 mA	↓
9	[3] TxD	↑
10	[3] RxD	↓
22	[3] CTS	↓
23	[3] RTS	↑

D-sub
25-pin Female

[1] Pin no 20 connected to pin no 21 internal in the terminal
[2] Only for E50, E100, E150
[3] Only for E50

RS-232
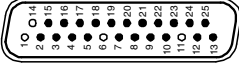
Pin no	Name	Signal direction Terminal<-> XXX
1	[4] +5 V > 200 mA	↓
2	TxD	↑
3	RxD	↓
5	0 V	
7	CTS	↓
8	RTS	↑
9		

D-sub
9-pin Male

[4] Only for E50 with 232-25-ADP. On all other terminals, not connected.

20.2.2   232-25-ADP adapter to E50
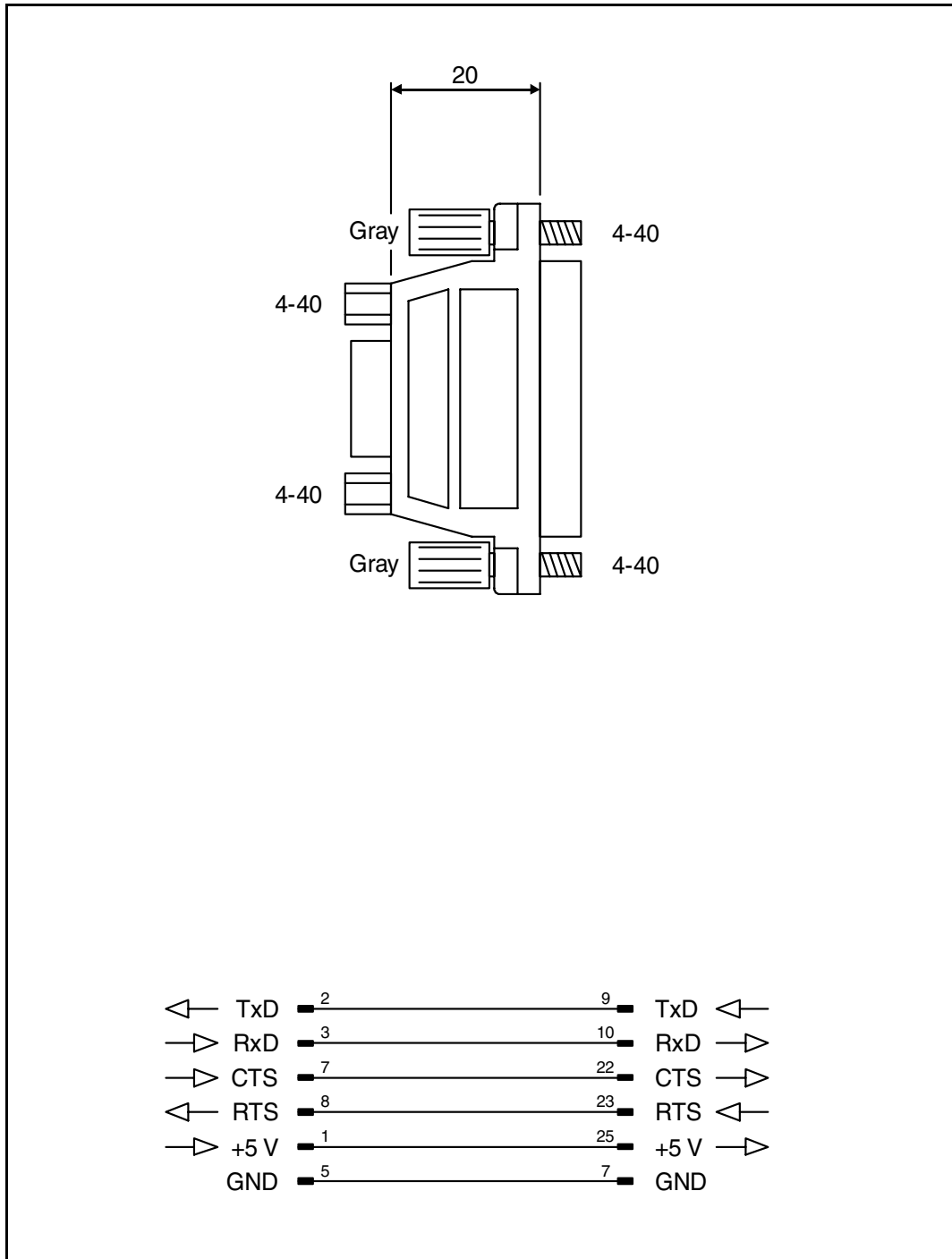
20.2.3   E50 Outline drawing
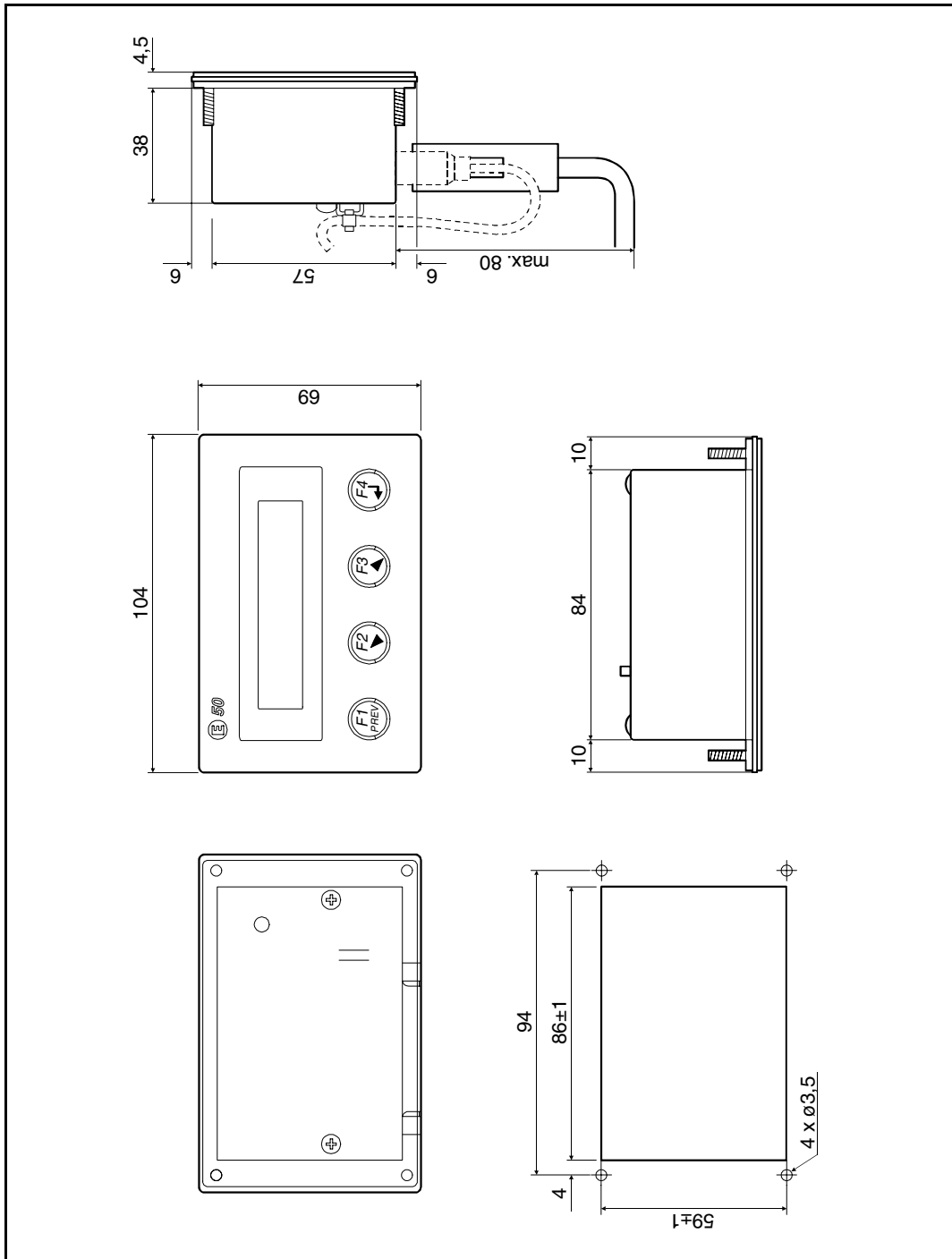
20.2.4   E100 Outline drawing
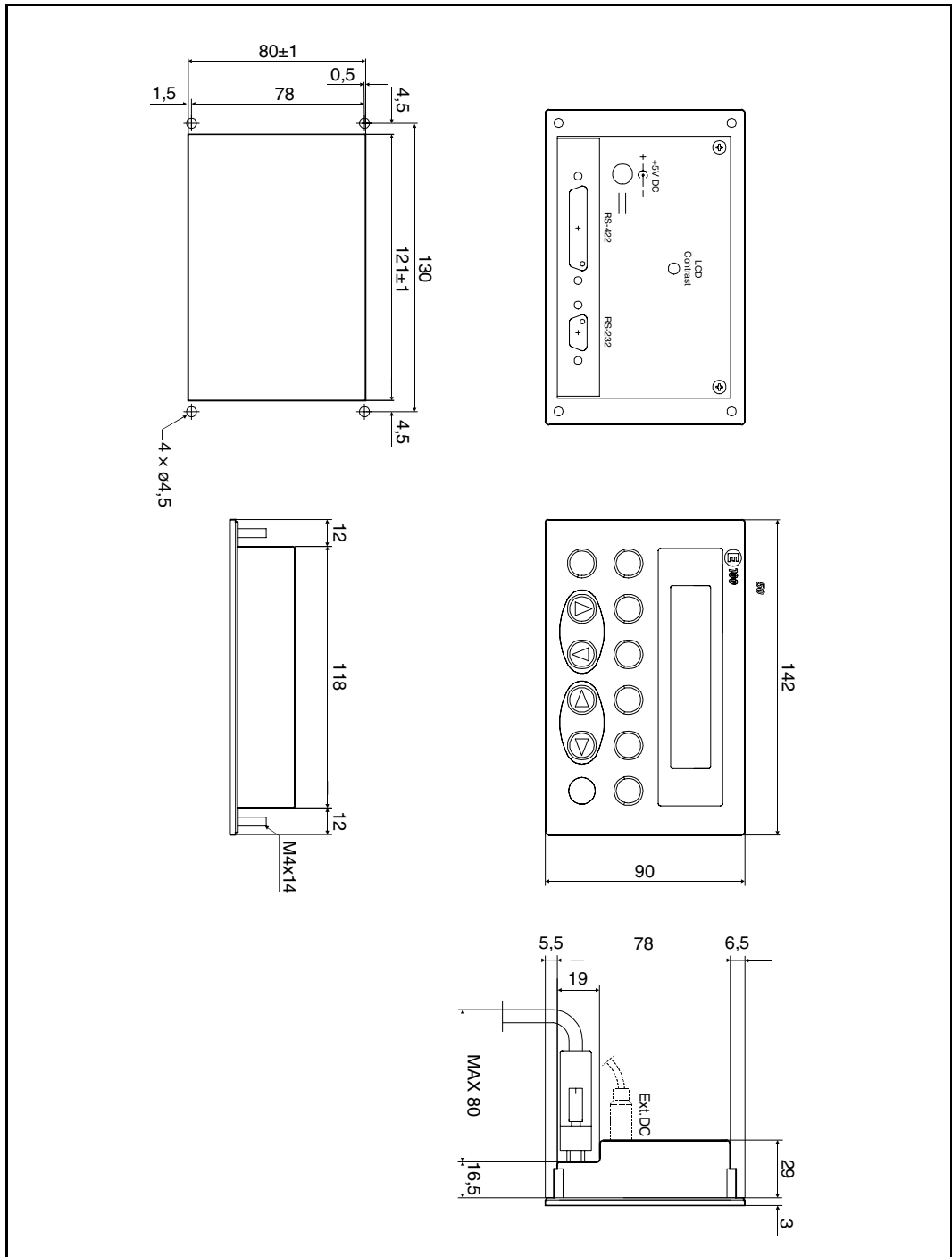
20.2.5   **E150 Outline drawing**
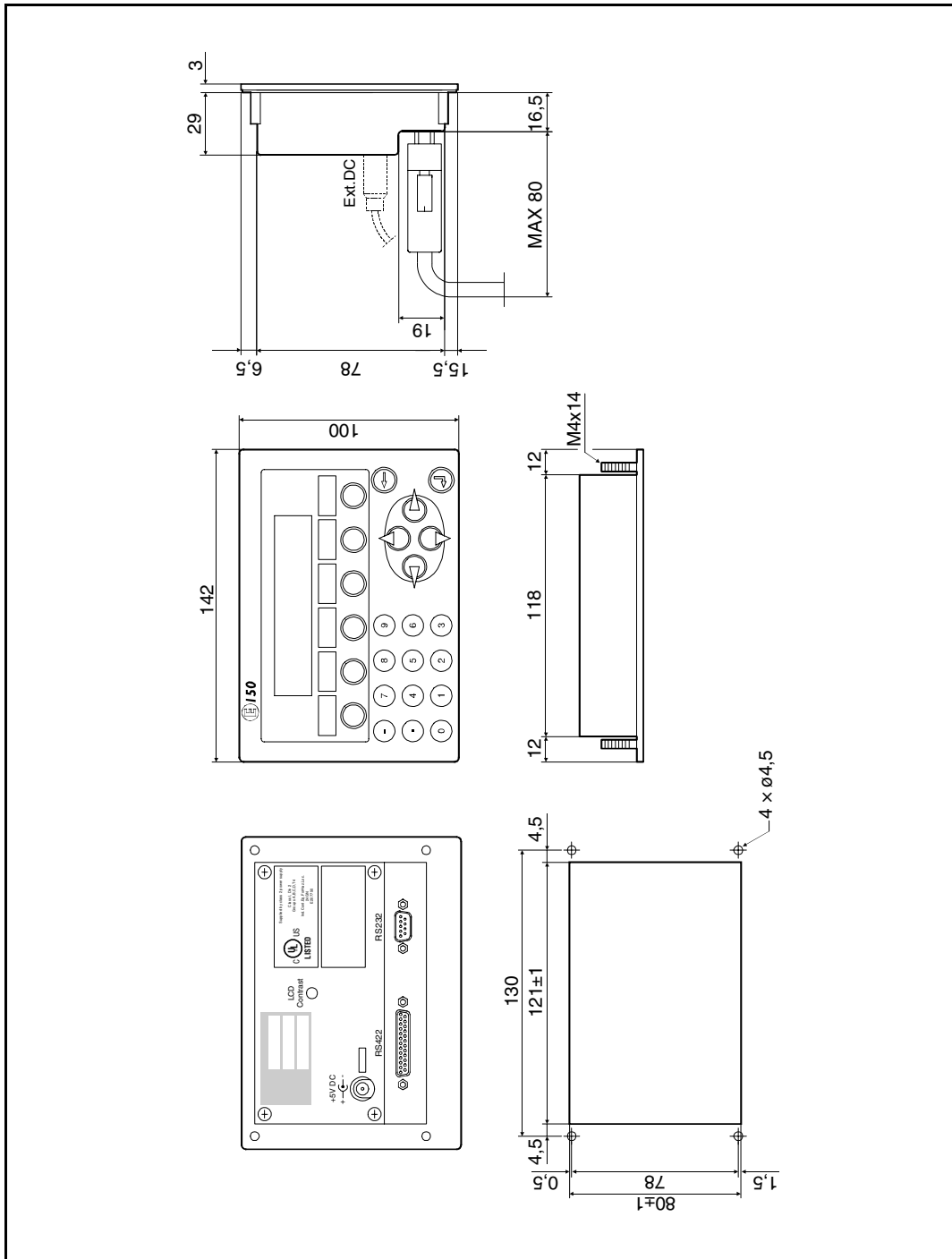
20.2.6   E150 Text strip

◆ MITSUBISHI ELECTRIC
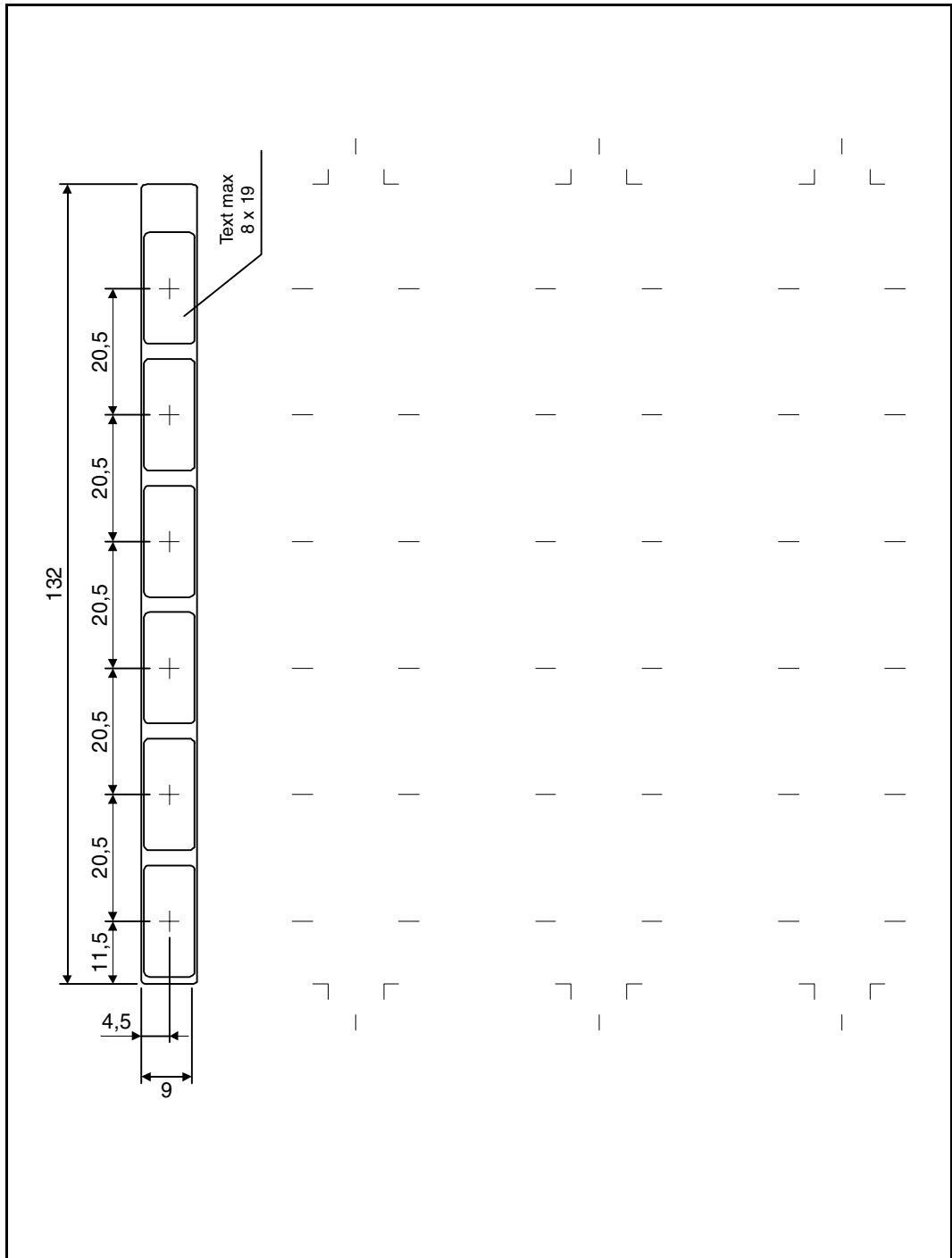
20.2.7   **E200 Outline drawing**
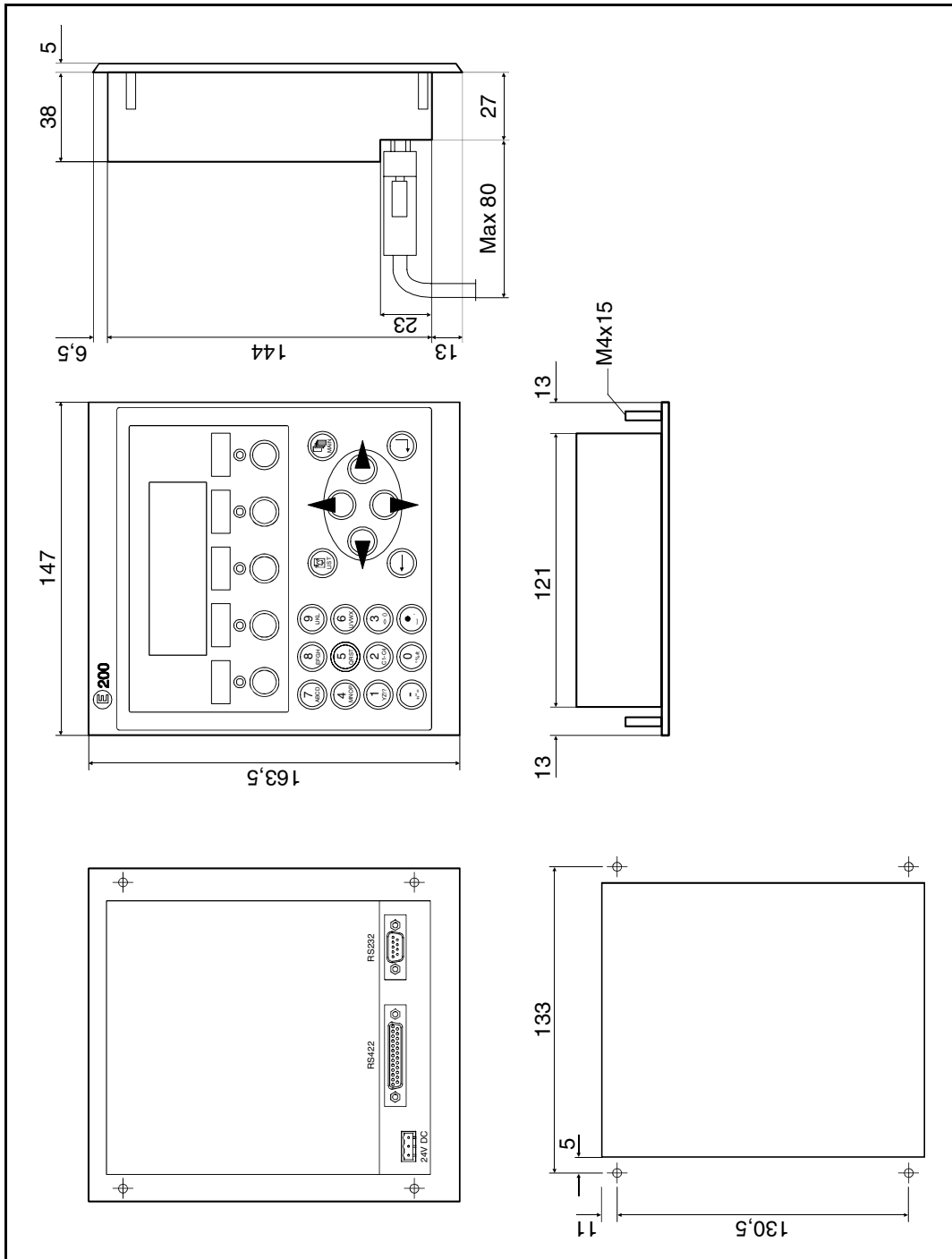
20.2.8   E200 Text strip

MITSUBISHI ELECTRIC
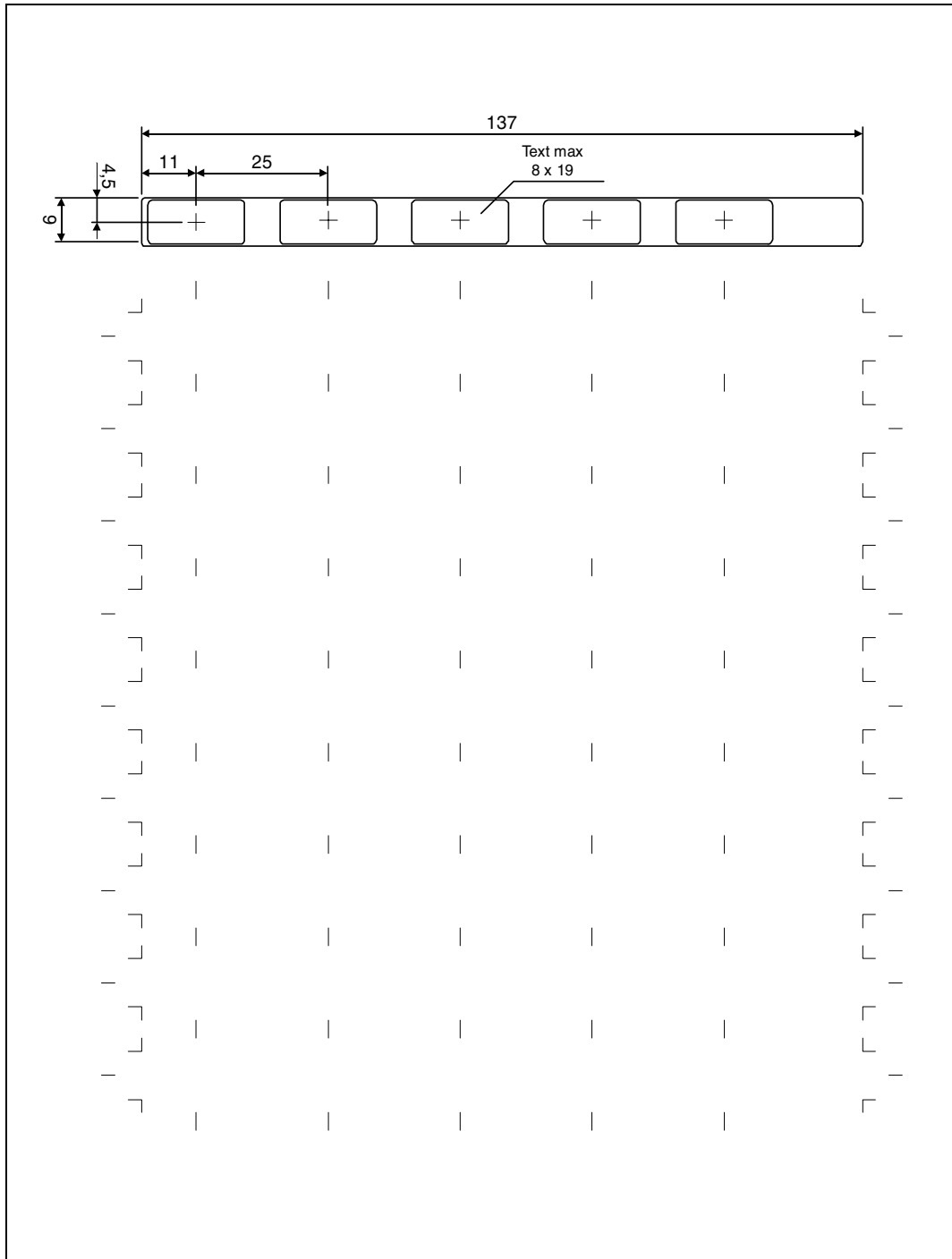
20.2.9   Programming cable RS232C
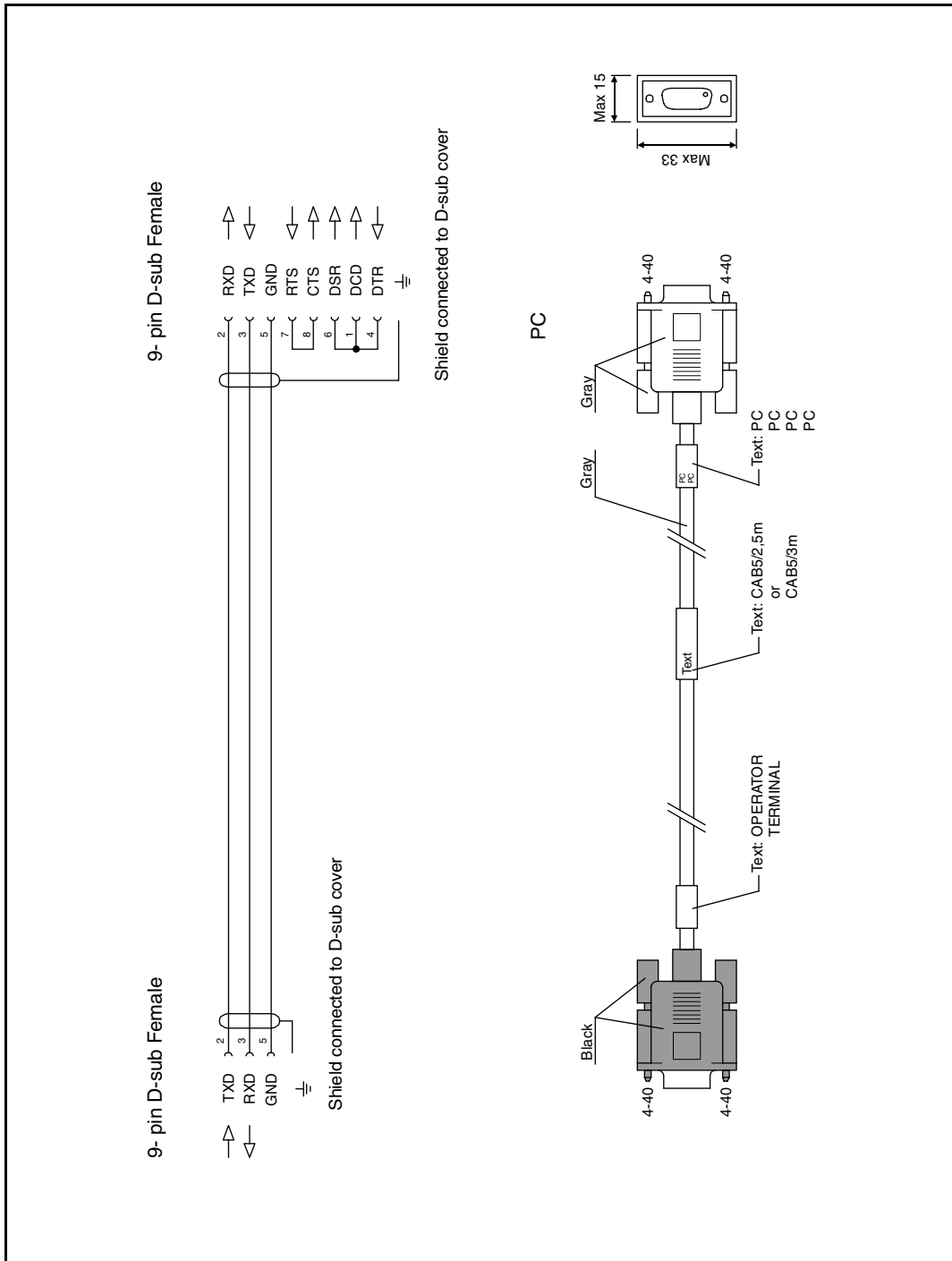
20.2.10 Programming cable RS422
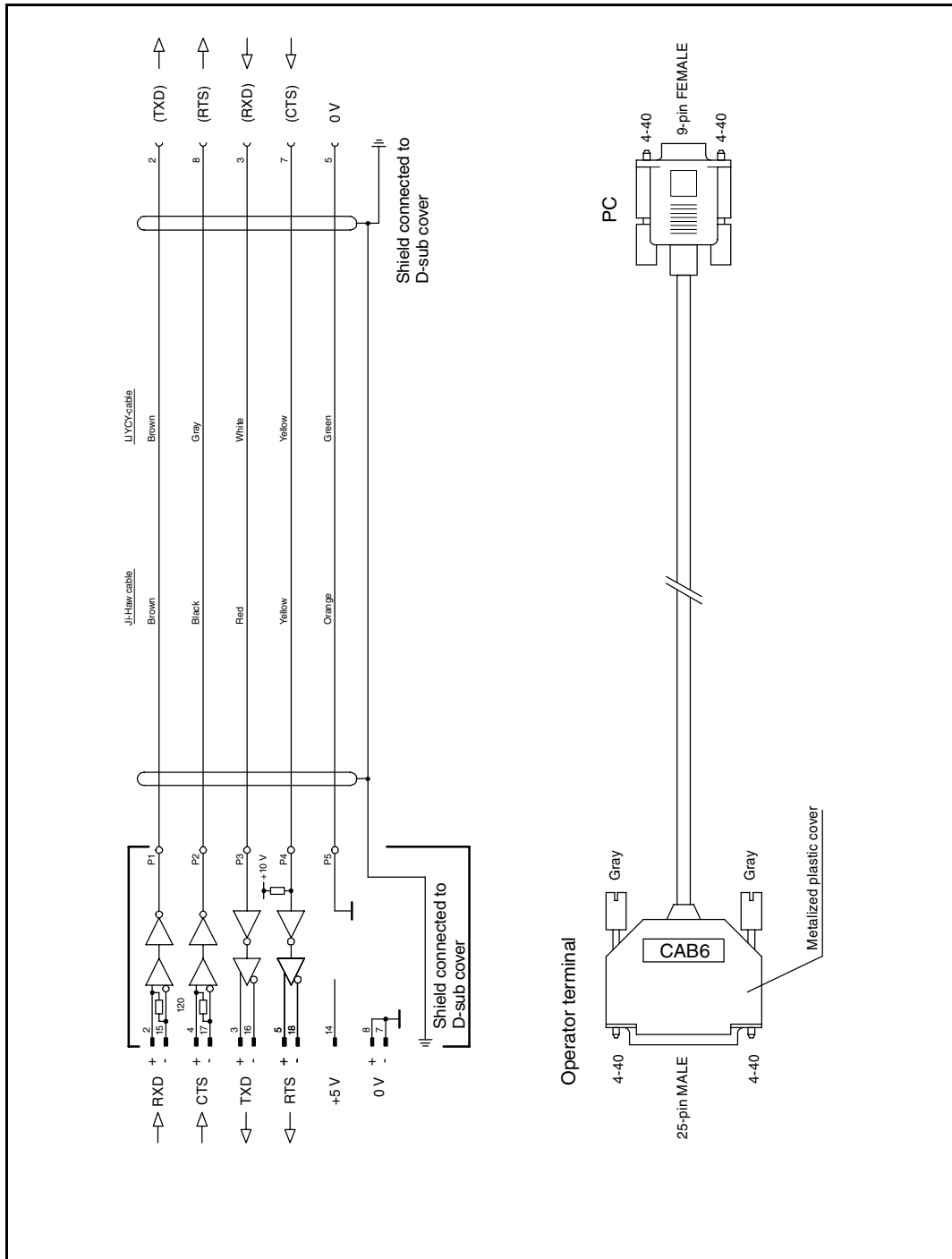
20.2.11   Symbol Chart for E100 and E150

20.2.12   Symbol Chart for E50 and E200

20.2.13   National characters for E50

The characters 33-125 in the table on the previous page (character table for E50 and E200) are also supported on E50. E50 also supports a number of national characters, depending on the selected national settings/character table.

Press the Alt key and enter a number from the table below on the numeric portion of your keyboard to view the character on the terminal.

Keyboard combination	Character table			
	Swedish	German	Norwegian/ Danish	French
Alt + 142	Å	Å	Å	
Alt + 143	Ä	Ä		
Alt + 153	Ö	Ö	Ö	
Alt + 134	å	å	å	
Alt + 132	ä	ä	ä	ä
Alt + 148	ö	ö	ö	ö
Alt + 129	ü	ü	ü	ü
Alt + 157	Ø	Ø	Ø	
Alt + 152		ÿ		
Alt + 154		Ü		
Alt + 155			ø	
Alt + 145			æ	
Alt + 144				É
Alt + 138				È
Alt + 136				Ê

Index

U

V

W

HEADQUARTERS

MITSUBISHI ELECTRIC **EUROPE**
EUROPE B.V.
German Branch
Gothaer Straße 8
D-92880 Ratingen
Phone: +49 (0) 2102 / 486-0
Fax: +49 (0) 2102 / 486-1120
e mail: megfamail@meg.mee.com

MITSUBISHI ELECTRIC **FRANCE**
EUROPE B.V.
French Branch
25, Boulevard des Bouvets
F-92741 Nanterre Cedex
Phone: +33 1 55 68 55 68
Fax: +33 1 55 68 56 85
e mail: factory.automation@fra.mee.com

MITSUBISHI ELECTRIC **IRELAND**
EUROPE B.V.
Irish Branch
Westgate Business Park, Ballymount
IRL-Dublin 24
Phone: +353 (0) 1 / 419 88 00
Fax: +353 (0) 1 / 419 88 90
e mail: sales.info@meir.mee.com

MITSUBISHI ELECTRIC **ITALY**
EUROPE B.V.
Italian Branch
Via Paracelso 12
I-20041 Agrate Brianza (MI)
Phone: +39 039 6053 1
Fax: +39 039 6053 312
e mail: factory.automation@it.mee.com

MITSUBISHI ELECTRIC **SPAIN**
EUROPE B.V.
Spanish Branch
Carretera de Rubí 76-80
E-08190 Sant Cugat del Vallés
Phone: +34 9 3 / 565 3131
Fax: +34 9 3 / 589 2948
e mail: industrial@sp.mee.com

MITSUBISHI ELECTRIC **UK**
EUROPE B.V.
UK Branch
Travellers Lane
GB-Hatfield Herts. AL10 8 XB
Phone: +44 (0) 1707 / 27 61 00
Fax: +44 (0) 1707 / 27 86 95
e mail: automation@meuk.mee.com

MITSUBISHI ELECTRIC **JAPAN**
CORPORATION
Office Tower "Z" 14 F
8-12,1 chome, Harumi Chuo-Ku
Tokyo 104-6212
Phone: +81 3 6221 6060
Fax: +81 3 6221 6075

MITSUBISHI ELECTRIC **USA**
AUTOMATION
500 Corporate Woods Parkway
Vernon Hills, IL 60061
Phone: +1 847 / 478 21 00
Fax: +1 847 / 478 22 83

MIDDLE EAST REPRESENTATIVE

Ilan & Gavish Ltd. **ISRAEL**
Automation Service
24 Shenkar St., Kiryat Arie
IL-49001 Petah-Tiqva
Phone: +972 (0) 3 / 922 18 24
Fax: +972 (0) 3 / 924 07 61
e mail: iandg@internet-zahav.net

EUROPEAN REPRESENTATIVES

GEVA **AUSTRIA**
Wiener Straße 89
AT-2500 Baden
Phone: +43 (0) 2252 / 85 55 20
Fax: +43 (0) 2252 / 488 60
e mail: office@geva.at

TEHNIKON **BELARUS**
Oktjabrskaya 16/5, Ap 704
BY-220030 Minsk
Phone: +375 (0)17 / 22 75 704
Fax: +375 (0)17 / 22 76 669
e mail: tehnikon@belsonet.net

Getronics b.v. **BELGIUM**
Control Systems
Pontbeeklaan 43
B-1731 Asse-Zellik
Phone: +32 (0) 2 / 467 17 51
Fax: +32 (0) 2 / 467 17 45
e mail: infoautomation@getronics.com

TELECON CO. **BULGARIA**
4, A. Ljapchev Blvd.
BG-1756 Sofia
Phone: +359 (0) 2 / 97 44 05 8
Fax: +359 (0) 2 / 97 44 06 1
e mail: —

INEA CR d.o.o. **CROATIA**
Losinjska 4 a
HR-10000 Zagreb
Phone: +385 (0) 1 / 36 940-01
Fax: +385 (0) 1 / 36 940-03
e mail: inea@inea.hr

AutoCont **CZECH REPUBLIC**
Control Systems s.r.o.
Nemocnicni 12
CZ-702 00 Ostrava 2
Phone: +420 59 / 6152 111
Fax: +420 59 / 6152 562
e mail: consys@autocont.cz

louis poulsen **DENMARK**
industri & automation
Geminivej 32
DK-2670 Greve
Phone: +45 (0) 70 / 10 15 35
Fax: +45 (0) 43 / 95 95 91
e mail: lpia@lpmail.com

UTU Elektrotehnika AS **ESTONIA**
Pärnu mnt.160i
EE-11317 Tallinn
Phone: +372 (0) 6 / 51 72 80
Fax: +372 (0) 6 / 51 72 88
e mail: utu@utu.ee

Beijer Electronics OY **FINLAND**
Ansatie 6a
FIN-01740 Vantaa
Phone: +358 (0) 9 / 886 77 500
Fax: +358 (0) 9 / 886 77 555
e mail: info@beijer.fi

UTECO A.B.E.E. **GREECE**
5, Mavrogenous Str.
GR-18542 Piraeus
Phone: +302 (0) 10 / 42 10 050
Fax: +302 (0) 10 / 42 12 033
e mail: sales@uteco.gr

Meltrade Automatika Kft. **HUNGARY**
55, Harmat St.
HU-1105 Budapest
Phone: +36 (0)1 / 2605 602
Fax: +36 (0)1 / 2605 602
e mail: office@meltrade.hu

SIA POWEL **LATVIA**
Lienes iela 28
LV-1009 Riga
Phone: +371 784 / 22 80
Fax: +371 784 / 22 81
e mail: utu@utu.lv

EUROPEAN REPRESENTATIVES

UAB UTU POWEL **LITHUANIA**
Savanoriu pr. 187
LT-2053 Vilnius
Phone: +370 (0) 52323-101
Fax: +370 (0) 52322-980
e mail: powel@utu.lt

INTEHSIS SRL **MOLDOVA**
Cuza-Voda 36/1-81
MD-2061 Chisinau
Phone: +373 (0)2 / 562 263
Fax: +373 (0)2 / 562 263
e mail: intehsis@mdl.net

Getronics b.v. **NETHERLANDS**
Control Systems
Donauweg 2 B
NL-1043 AJ Amsterdam
Phone: +31 (0) 20 / 587 67 00
Fax: +31 (0) 20 / 587 68 39
e mail: info.gia@getronics.com

Beijer Electronics AS **NORWAY**
Teglverksveien 1
N-3002 Drammen
Phone: +47 (0) 32 / 24 30 00
Fax: +47 (0) 32 / 84 85 77
e mail: info@beijer.no

MPL Technology Sp. z o.o. **POLAND**
ul. Sliczna 36
PL-31-444 Kraków
Phone: +48 (0) 12 / 632 28 85
Fax: +48 (0) 12 / 632 47 82
e mail: krakow@mpl.pl

Sirius Trading & Services srl **ROMANIA**
Str. Biharia No. 67-77
RO-013981 Bucuresti 1
Phone: +40 (0) 21 / 201 1146
Fax: +40 (0) 21 / 201 1148
e mail: sirius@siriustrading.ro

INEA d.o.o. **SLOVENIA**
Stegne 11
SI-1000 Ljubljana
Phone: +386 (0) 1-513 8100
Fax: +386 (0) 1-513 8170
e mail: inea@inea.si

Beijer Electronics AB **SWEDEN**
Box 426
S-20124 Malmö
Phone: +46 (0) 40 / 35 86 00
Fax: +46 (0) 40 / 35 86 02
e mail: info@beijer.se

ECONOTEC AG **SWITZERLAND**
Postfach 282
CH-8309 Nürensdorf
Phone: +41 (0) 1 / 838 48 11
Fax: +41 (0) 1 / 838 48 12
e mail: info@econotec.ch

GTS **TURKEY**
Darülaceze Cad. No. 43 Kat. 2
TR-80270 Okmeydani-Istanbul
Phone: +90 (0) 212 / 320 1640
Fax: +90 (0) 212 / 320 1649
e mail: gts@turk.net

CSC Automation Ltd. **UKRAINE**
15, M. Raskova St., Fl. 10, Office 1010
UA-02002 Kiev
Phone: +380 (0) 44 / 238-83-16
Fax: +380 (0) 44 / 238-83-17
e mail: csc-a@csc-a.kiev.ua

EURASIAN REPRESENTATIVES

Avtomatika Sever Ltd. **RUSSIA**
Lva Tolstogo St. 7, Off. 311
RU-197376 St Petersburg
Phone: +7 812 / 11 83 238
Fax: +7 812 / 11 83 239
e mail: as@avtsev.spb.ru

CONSYS **RUSSIA**
Promyshlennaya St. 42
RU-198099 St Petersburg
Phone: +7 812 / 325 36 53
Fax: +7 812 / 147 20 55
e mail: consys@consys.spb.ru

Electrotechnical **RUSSIA**
Systems Siberia
Partizanskaya St. 27, Office 306
RU-121355 Moscow
Phone: +7 095 / 416-4321
Fax: +7 095 / 416-4321
e mail: info@eltechsystems.ru

Electrotechnical **RUSSIA**
Systems Siberia
Shetinkina St. 33, Office 116
RU-630088 Novosibirsk
Phone: +7 3832 / 22-03-05
Fax: +7 3832 / 22-03-05
e mail: info@eltechsystems.ru

Elektrostyle **RUSSIA**
ul. Garschina 11
RU-140070 Moscow
Phone: +7 095 / 514 9316
Fax: +7 095 / 514 9317
e mail: info@estl.ru

Elektrostyle **RUSSIA**
Krasnij Prospekt 220-1
Office No. 312
RU-630049 Novosibirsk
Phone: +7 3832 / 10 66 18
Fax: +7 3832 / 10 66 26
e mail: info@estl.ru

ICOS **RUSSIA**
Industrial Computer Systems Zao
Ryazanskij Prospekt 8a, Office 100
RU-109428 Moscow
Phone: +7 095 / 232 - 0207
Fax: +7 095 / 232 - 0327
e mail: mail@icos.ru

NPP Uralelektra **RUSSIA**
ul. Sverdlova 11a
RU-620027 Ekaterinburg
Phone: +7 34 32 / 53 27 45
Fax: +7 34 32 / 53 27 45
e mail: elektra@etel.ru

SSMP Rosgidromontazh Ltd. **RUSSIA**
23, Lesoparkovaya Str.
RU-344041 Rostov On Don
Phone: +7 8632 / 36 00 22
Fax: +7 8632 / 36 00 26
e mail: —

STC Drive Technique **RUSSIA**
ul. Bajkalskaja 239, Office 2 - 23
RU-664075 Irkutsk
Phone: +7 3952 / 24 38 16
Fax: +7 3952 / 23 02 98
e mail: privod@irk.ru

STC Drive Technique **RUSSIA**
Poslannikov Per. 9, str.1
RU-107005 Moscow
Phone: +7 095 / 790-72-10
Fax: +7 095 / 790-72-12
e mail: info@privod.ru

AFRICAN REPRESENTATIVE

CBI Ltd. **SOUTH AFRICA**
Private Bag 2016
ZA-1600 Isando
Phone: +27 (0) 11/ 928 2000
Fax: +27 (0) 11/ 392 2354
e mail: cbi@cbi.co.za

MITSUBISHI ELECTRIC INDUSTRIAL AUTOMATION

Gothaer Straße 8	Phone: +49 2102 486-0	Fax: +49 2102 486-7170	www.mitsubishi-automation.de
D-40880 Ratingen	Hotline: +49 1805 000-765	megfa-mail@meg.mee.com	www.mitsubishi-automation.com