

OMRON ELECTRONICS EUROPE

MiniLSS V1.0

USER MANUAL

omron

SECTION 1 General	1
1.1. Hardware Requirements	1
1.2. Installation	1
SECTION 2 Getting Started	5
2.1. The Editor Screen	5
2.2. The Editor Screen in LSS Mode	6
2.3. Editing Ladder Diagrams	7
2.4. Editing in MiniLSS Mode	7
2.5. Editing in LSS Mode	9
2.6. Editing Instruction Lists	11
2.7. Editing in Full Screen Instruction List	14
2.8. Editing Address Symbols and Comments	15
2.9. Editing Diagram Comments	17
2.10. Line Editing Commands	18
SECTION 3 Programming	20
3.1. Files	20
3.2. Designing the diagrams	20
3.3. Using symbols	21
3.4. Using the clipboard	21
3.4.1. Copying, cutting and pasting diagram portions	21
3.4.2. Copying, cutting and pasting instruction lines	23
3.4.3. Clearing	25
3.4.4. Converting program format with the clipboard	25
3.4.5. Saving the clipboard to a file	28
3.4.6. Loading the clipboard from a file	29
3.5. Using the Address Exchange Editor	29
SECTION 4 Printing Documents	32
4.1. Choosing the printer	32
4.2. Orientation	32
4.3. Page Length	35
4.4. Headers	35
4.4.1. Listing Codes in a Header File	35
4.4.2. Header Alignments	36
SECTION 5 Inside Menus	38
5.1. Ladder Editor Menus, MiniLSS Mode	38
5.1.1. Files	38
5.1.2. The Edit-Menu	39
5.1.3. Diagram Elements	40
5.1.4. Searching	41
5.1.5. Compiling	41
5.1.6. PLC Utilities	41
5.1.7. Symbols	42
5.1.8. Options	42
5.2. Ladder Editor Menus, LSS Mode	44
5.2.1. Programming	44
5.2.2. Search	45
5.2.3. When PLC Connection is ON	45
5.3. Symbol Table Editor menus, MiniLSS Mode	46
5.3.1. Edit Menu	46
5.3.2. Goto Menu	46
5.3.3. Search Menu	46
5.4. Symbol Table Editor menus, LSS Mode	47
5.4.1. Search submenu	48
5.5. File management menu, LSS Mode	48

5.6.	Setup-menu, LSS Mode	49
SECTION 6 Online Properties		51
6.1.	Monitoring a Diagram.....	51
6.2.	Monitoring the Symbol Table.....	53
6.3.	Multimonitoring Pages.....	53
6.3.1.	Structure of a Multimonitoring Page.....	54
6.3.2.	Editing a Multimonitoring Page.....	55
6.3.3.	Changing PLC states in the Multimonitoring Page.....	56
6.3.4.	Setting LR Area Size	56
6.3.5.	Setting Filter Values	57
6.3.6.	Other PLC Commands.....	57
6.4.	Communication Errors.....	58
SECTION 7 Compiling		59
7.1.	Compiling Diagrams.....	59
7.2.	Compiling Object Code to Diagrams	61
7.3.	Comparing Programs	61
SECTION 8 Customizing Display Colors.....		63

MiniLSS is a programming software for SP-series PLCs. With MiniLSS you can design the program, transfer it to and from a PLC, monitor PLC statuses and print your program documents.

1.1. Hardware Requirements

MiniLSS requires the following hardware:

- IBM compatible personal computer (IBM AT, 386 or 486)
- Hard Disk
- Minimum of 640 kbytes memory
- 1.44 Mb diskette drive

The following is required to download the program to a PLC:

- Serial Port (RS-232C)
- SP <-> PC cable

The following is required to print documents:

- Parallel port
- HP LaserJet compatible laser printer or IBM Graphics compatible matrix printer

PLC types supported by MiniLSS are SP10, SP16 and SP20.

1.2. Installation

MiniLSS is delivered in one 1.44 Mb diskette. To install the software, follow the following steps:

- Put the MiniLSS diskette to your diskette drive (A: or B:)
- Type: A:INSTALL (If your diskette drive is A:)

The installation program screen opens as illustrated below:

MiniLSS is a programming software for SP-series PLCs. With MiniLSS you can design the program, transfer it to and from a PLC, monitor PLC statuses and print your program documents.

1.1. Hardware Requirements

MiniLSS requires the following hardware:

- IBM compatible personal computer (IBM AT, 386 or 486)
- Hard Disk
- Minimum of 640 kbytes memory
- 1.44 Mb diskette drive

The following is required to download the program to a PLC:

- Serial Port (RS-232C)
- SP <-> PC cable

The following is required to print documents:

- Parallel port
- HP LaserJet compatible laser printer or IBM Graphics compatible matrix printer

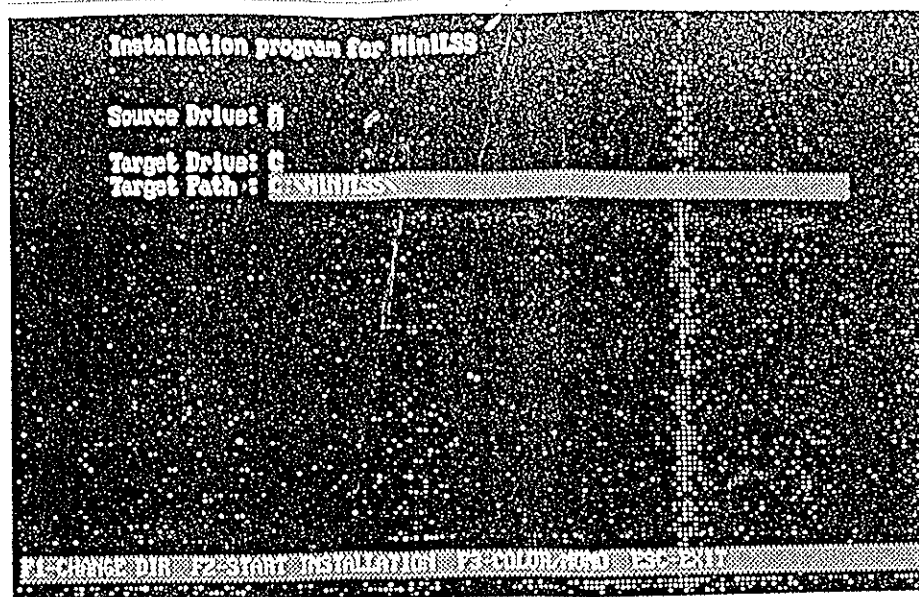
PLC types supported by MiniLSS are SP10, SP16 and SP20.

1.2. Installation

MiniLSS is delivered in one 1.44 Mb diskette. To install the software, follow the following steps:

- Put the MiniLSS diskette to your diskette drive (A: or B:)
- Type: A:INSTALL (If your diskette drive is A:)

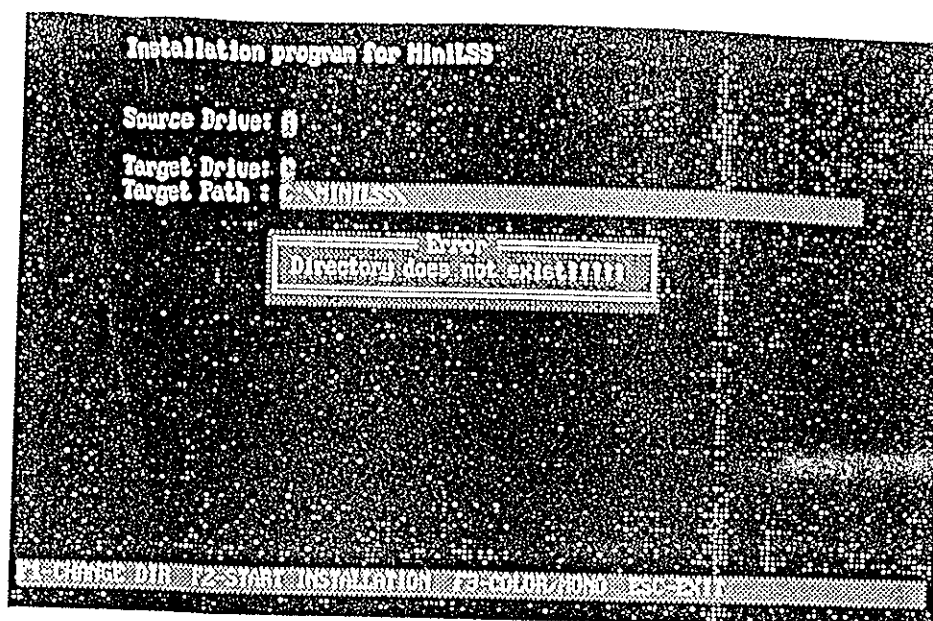
The installation program screen opens as illustrated below:



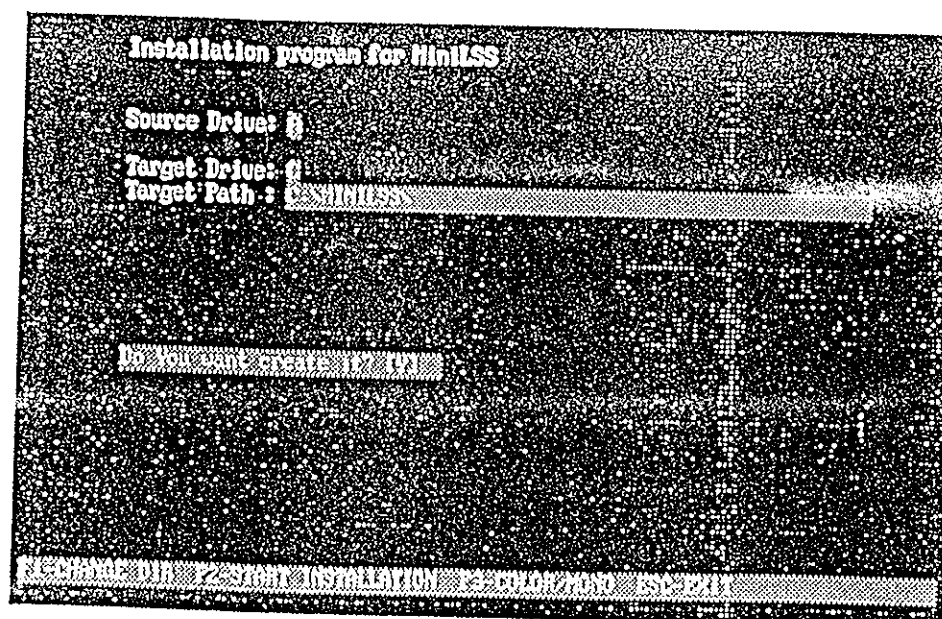
If you want to install MiniLSS to different directory, press **[F1]** and select the desired directory.

SECTION 1

If the named directory does not exist, the installation program will display a message



If you got the message, press **[Esc]**. Installation program will then ask if you want to create such a directory:



Just press **[Enter]** to accept the new directory

To start the installation press **[F2]**


Minil SS will now be copied to your hard disk. Store the original diskette in a safe place. If you would also like to use MinilSS from other directories, again

from the C:\WINILSS directory, you should modify your PATH statement in the AUTOEXEC.BAT file. Consult your DOS manual for doing that.


SECTION 2

Getting Started

If you didn't modify your PATH statement, you must change to the MiniLSS directory before starting the program.
Start the program by typing:

MINILSS 

If you are using MiniLSS with monochrome display, you can select the monochrome display attributes by starting MiniLSS with command line parameter "MONO", like:

MINILSS MONO 

The MiniLSS outlook can be selected to a LSS-style outlook (called LSS Mode or LSS-look in this manual) or to its own outlook (called MiniLSS Mode).

2.1.The Editor Screen

MiniLSS starts with loading run time files. After file loading is completed, the editor screen opens, as in the following picture:



Note! If you have selected the LSS-Look option, the screen will appear different.

At the top line of the screen is the menu bar. Below this is the diagram area, where you will put your relay symbols and instructions. Just below the diagram area are three lines (initialized with MINILSS text) which are reserved for displaying information about contacts at the cursor position. Below these lines is the command key guideline, which gives function key declarations. The bottom line is the status line, which shows available memory in the computer, the off-line/on-line status, addressing mode and selected PLC type.

The cursor is the white block in the top left corner of the diagram area. If the cursor is visible, you are working in the diagram area; if not, the focus is either in the menu bar or in the instruction list (discussed later). To switch between menu bar and the diagram area press **[Esc]** or hold the **[Alt]** key down and press any of the highlighted character keys.

Online help can be accessed with **[F1]**

2.2.The Editor Screen in LSS Mode

In LSS Mode, the editor screen appears as in the following picture:



The topline contains the editor mode (WRITE/READ). Below it begins the diagram space. Just below the diagram space is the edit line, where ladder elements and their addresses are edited. The line below the edit line is the function key guide, and the bottom line is the status line, which shows available memory in the computer, the off-line/on-line status, addressing mode and selected PLC type.

To go to menu press **[Ctrl][M]** or **[End]**



Online help can be accessed with **[Ctrl][H]**


2.3.Editing Ladder Diagrams

Relay ladder elements are



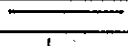

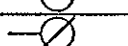

- Normally open contact
- Normally closed contact
- Output coil
- Negated output coil
- Horizontal short
- Vertical short

2.4.Editing in MiniLSS Mode


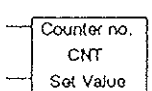
The elements can be accessed from element-menu or with shortcut keys. Select the element by positioning the cursor and then select the menu item or use a command key. The element will appear at the cursor position. If the element is other than a vertical or horizontal short, assign an address to the element. Type in the address and press . You can also leave the address blank by just pressing  and specify the address afterwards.

To enter or change address of an existing contact, position the cursor on the contact and press <space>. To enter or change operands to an existing function, counter or timer, position the cursor on the operand and press .

Shortcut function keys for relay elements are shown on the function key guideline. Additional keys can be used as well. The keys and corresponding elements are:

Element name	Element symbol	Shortcut key
Normally open contact		F2, "
Normally closed contact		F3, /
Horizontal short		F4, -
Vertical short		F5, !,
Output coil		F6, O
Negated output coil		F7

In addition to basic elements there are timers, counters and functions. Their representation and shortcut keys are:

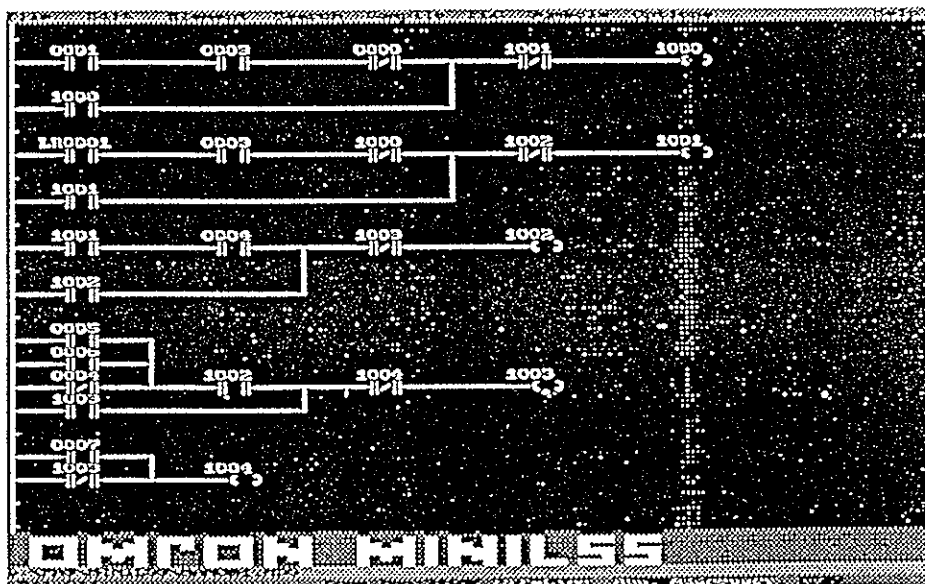
Element name	Element symbol	Shortcut key
Timer		F9
Counter		F10

Function	<div data-bbox="954 219 1086 331"> Function name Operand 1 Operand 2 Operand 3 </div>	F8, F
----------	--	-------

The elements can be drawn in any order. Their addresses and operands can be filled at any time; the correct syntax is required only when compiling. In the positioning of elements the following rules must be obeyed:

- If there is output, timer, counter or function in a line, nothing can be inserted to the right of it.
- If there are contacts or shorts in a line, no outputs, timers, counters or functions can be inserted to the left of them.
- Elements cannot overlap.

The ladder diagram appears as follows:



Elements can be deleted with the **[Del]** key. Position the cursor on the required contact and press **[Del]**. Function blocks, counters and timers are deleted by positioning the cursor over top of them and pressing **[Del]**.

Elements can also be deleted with the backspace key. Position the cursor to the right of the contact to be deleted and press **[←]**. The element is deleted and the cursor moves one column to the left.

Vertical shorts cannot be deleted with the **[Del]** key. Instead they are toggled, i.e. drawing another vertical short will delete the existing one.

Empty lines can be deleted by holding the **[Ctrl]** key down and pressing **[Y]**. Empty lines can be inserted by pressing **[Ins]**.

To move elements in one circuit rightwards starting at cursor location hold the **[Ctrl]** key down and press **[→]**.

To move elements leftwards hold the **[Ctrl]** key down and press **[←]**.

Outputs, timer, counters and functions can be forced to appear right-justified. Select the *Justified Outputs* from *Options* menu. The justification affects only the display; it do not change the actual positions of elements. The horizontal line needed to connect the justified outputs is displayed as dashed line to indicate that it does not consist of true horizontal line elements. When you select the *Justified Outputs* again, the original positions are restored.

2.5.Editing in LSS Mode

In LSS Mode the editor is either in WRITE-mode, READ-mode or in DELETE-mode. Diagrams can be edited only in WRITE-mode. In READ-mode the diagrams can be viewed and searched, but not altered. Set the editor to WRITE-mode with key **[F2]**, and to READ-mode with key **[F1]**.

DELETE-mode can be accessed with key **[↑] [F1]** (Shift + F1).

When switching to READ-mode, the diagrams must be syntactically correct, since a compilation to PLC object code takes place. In READ-mode the corresponding line numbers in the PLC are displayed.

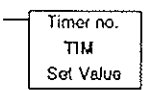
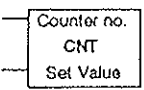
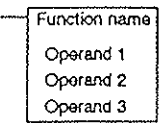
The elements can be accessed with function keys. Position the cursor to a place where the element should appear and use a command key to select the element type. The element will appear in the edit line. Enter the address if required. Use function keys to select the memory area (e.g. LR). When the element and its address is as desired hit **[Enter]** and the element will be placed into the cursor's position.

To change address of an existing contact position the cursor on the contact, its type and current address will be displayed in the edit line. Change the address and press **[Enter]**. To enter or change operands to an existing function, counter or timer, position the cursor on the required operand, type the address into the edit line and press **[Enter]**.

Shortcut function keys for relay elements are shown on the function key guideline. The keys and corresponding elements are:

Element name	Element symbol	Function key
Normally open contact		F4,F5
Normally closed contact		Shift+F4, Shift+F5
Horizontal short		F8
Vertical short		F6
Output coil		F7
Negated output coil		Shift+F7

In addition to basic elements there are timers, counters and functions. Their representation and function keys are:


Element name	Element symbol	Function key
Timer		Ctrl+F9
Counter		Ctrl+F10
Function		F10



To set the PLC memory areas, following keys are used:



Memory Area	Function key
LR	Ctrl+F1
DR	Ctrl+F2
*DR	Ctrl+F3
# (Constant)	Ctrl+F4
T/C	Ctrl+F5
I/O, Work Area	Ctrl+F6

The elements can be drawn in any order; the correct syntax is required only when compiling. In the positioning of elements the following rules must be obeyed:

- If there is output, timer, counter or function in a line, nothing can be inserted to the right of it.
- If there are contacts or shorts in a line, no outputs, timers, counters or functions can be inserted to the left of them.
- Elements cannot overlap.

Elements can be deleted with the backspace key. Position the cursor to the right of the contact to be deleted and press . The element is deleted and the cursor moves one column to the left.

To delete vertical short press first  so that the vertical short appears in the edit line and then hit .

Empty lines can be deleted by holding the  key down and pressing .

Empty lines can be inserted by pressing .

Elements can also be deleted in DELETE-mode. Set the DELETE-mode by pressing **[↑]** **[F1]** (Shift + F1). In DELETE-mode, the element under the cursor is deleted by pressing **[Enter]**. After the element is deleted the cursor will move one column to the right.

To move elements in one circuit rightwards starting at cursor location hold the **[Ctrl]** key down and press **[→]**.

To move elements leftwards hold the **[Ctrl]** key down and press **[←]**.

Outputs, timer, counters and functions can be forced to appear right-justified. Select the *Justified Outputs* from *Setup* menu. The justification affects only the display; it does not change the actual positions of elements. The horizontal line needed to connect the justified outputs is displayed as dashed line to indicate that it does not consist of true horizontal line elements. When you select the *Justified Outputs* again, the original positions are restored.

2.6.Editing Instruction Lists

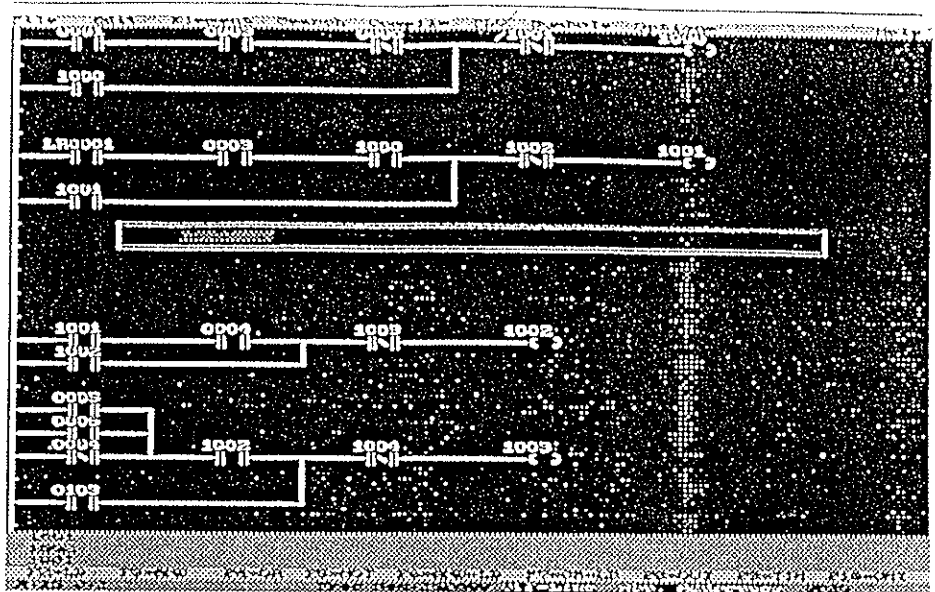
Note! Instruction list elements are not supported in LSS Model

Instruction list and ladder diagrams can be combined in the Diagram Editor. You can e.g. design the pure combination logic portions with ladder diagrams and function emphasized portions with instruction list. Of course, you may design the entire program with either ladder diagrams or instruction lists.

An instruction list program is divided into one or more instruction list blocks, with one instruction list element as the smallest block. Dividing the list into smaller blocks will help you to structure your program, thus making it much more readable. Although you are not required to divide the program, this method is strongly recommended. Diagram comments (discussed later) cannot be inserted into middle of the instruction list unit; instead they must be in between the list blocks (or list elements).

An instruction list element is inserted into the diagram space either by selecting it from the menu or by holding the **[Ctrl]** key down and pressing **[F1]**


Place the cursor in the line where you want the instruction list to begin and insert the instruction list element. The screen appears as follows:



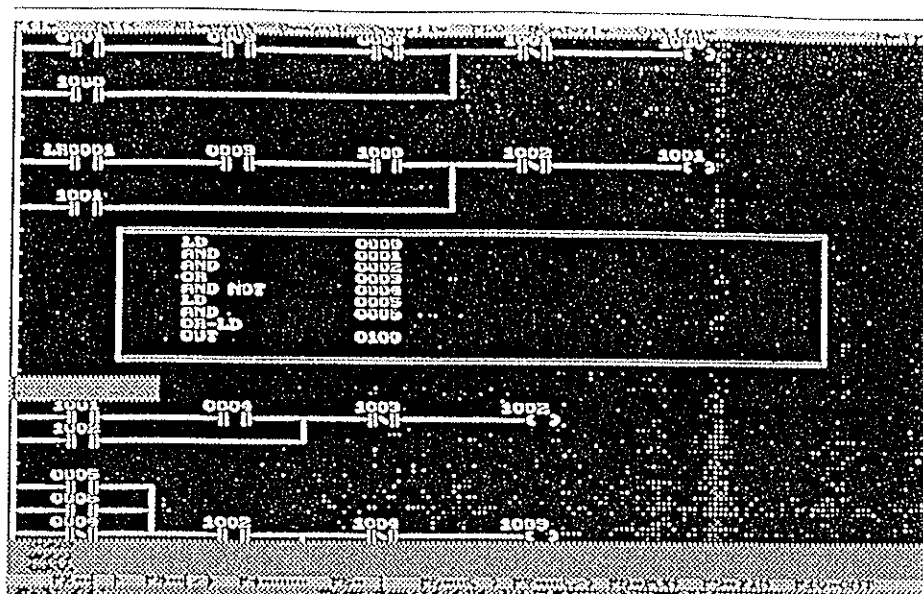
The empty frame in the middle of the screen is a placeholder for the instruction list. The ladder editor cursor will disappear and a normal text cursor will appear. The instruction field of the first program line is highlighted. The function key definitions will change; these changes will be seen on the command key guideline.

The length of the instruction list element is not limited; the frame will grow and shrink with the insertion or deletion of instructions. Instructions can be entered with function keys, as displayed on the command key guideline, or they may also be entered by typing their names. When typing names, case is not important.

Almost every instruction will include both an instruction code field and an instruction operand field. The instruction code must be entered first, since the number of operands depends on the instruction used. Type in the instruction

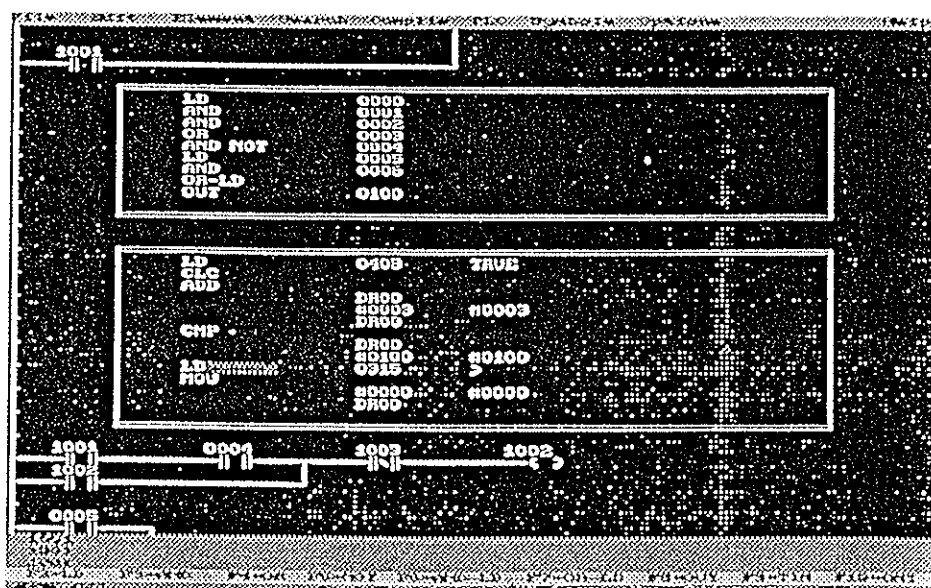
and press , or use the function keys. After the instruction code field is filled in, the cursor moves to the next field, which in most cases is the operand field (if the instruction does not include operands, the cursor moves to the next instruction's instruction code field). Fill in the operand(s) and then type in the next instruction code. After several instructions, the screen will appear as follows:

SECTION 2



You can insert several instruction list elements. It is best to program every logical entity as one element, as this will help you later when trying to locate certain program units.

The instruction list is divided into three columns: the leftmost column is the instruction code column, the middle column is the operand/address column and the rightmost column is the symbol column. In the previous picture there are no symbols defined, so the rightmost column is empty. The instruction lists in the following picture include some addresses that have predefined symbols, so the third column is used:



You can move from the address/operand field to the symbol field with **[F10]** (tab) or by moving the cursor beyond the right end of address field. To move from the symbol field back to the address field, hold the shift key down and press **[F10]**, or move the cursor beyond the left end of symbol field. Press the

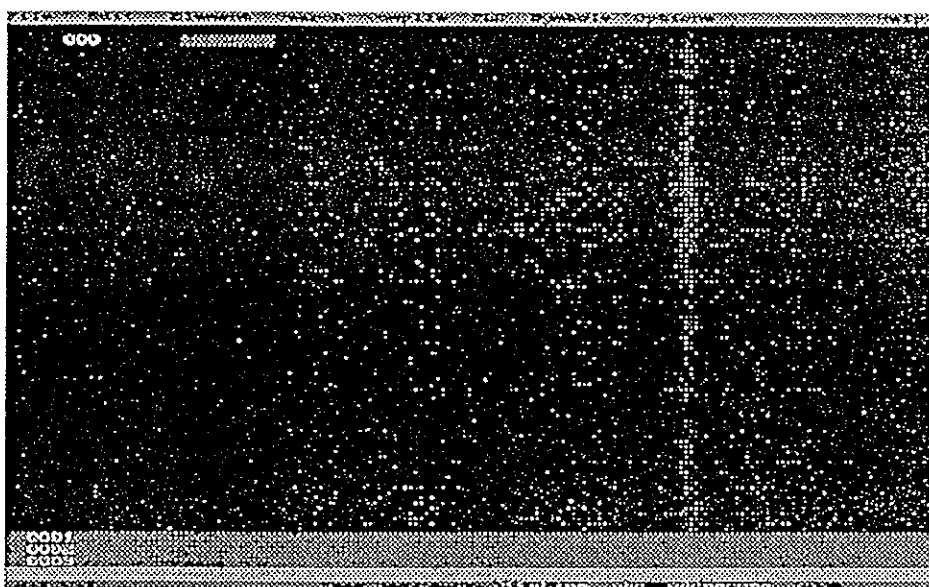
↑ and ↓ keys to move from one instruction to another. At the end of every instruction list element there is always an empty instruction. When the cursor is there and an instruction is typed, this instruction is automatically inserted at the end of the list and an empty instruction added to the end of the list element.

To insert a new instruction into the middle of the list, move the cursor (and the highlight) to the line where you want the new instruction to be placed and then press **Ins**. To delete an entire instruction hold the **Ctrl** key down and press **Y**.

When you move upwards from the top instruction or downwards from the last instruction, the ladder cursor appears and the command guideline changes, indicating that you are back in the ladder area.

2.7.Editing in Full Screen Instruction List

Apart from the diagrams, the PLC program can be edited also in instruction list editor. Select the *Change Editor* selection in *Edit* menu. If there are diagrams in the working space, they are compiled to one instruction list and the full screen instruction editor appears. The editor is illustrated below:

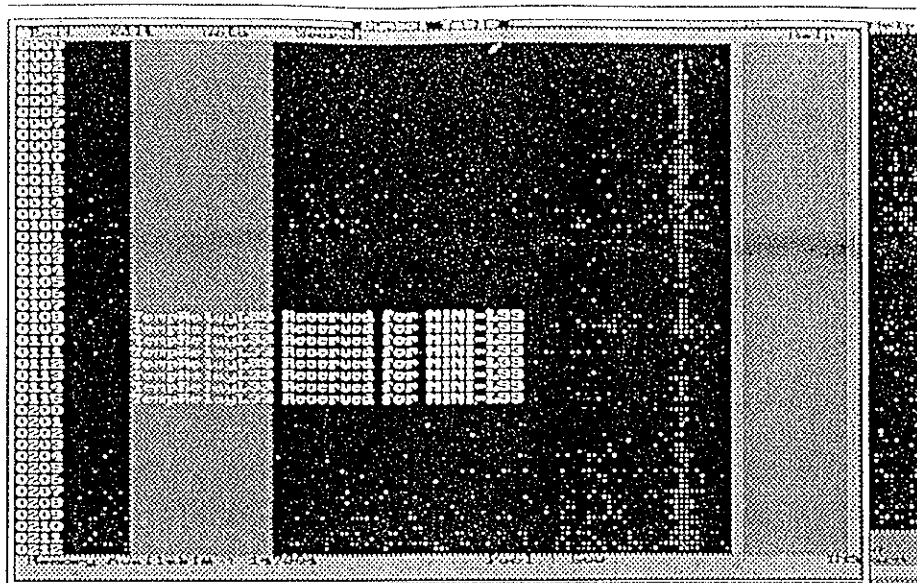


Almost every instruction will include both an instruction code field and an instruction operand field. The instruction code must be entered first, since the number of operands depends on the instruction used. Type in the instruction

and press **Enter**, or use the function keys. After the instruction code field is filled in, the cursor moves to the next field, which in most cases is the operand field (if the instruction does not include operands, the cursor moves to the next instruction's instruction code field). Fill in the operand(s) and then type in the next instruction code. After several instructions, the screen will appear as follows:

1965

$$x^2 + 2x + 1 = (x+1)^2$$



The topmost line is the menu bar. In LSS Mode the menu bar is absent, and the menu can be called with **[Ctrl][M]** or **[End]**.

The screen is divided into four columns. The leftmost column is the address column. This is always automatically filled in by MiniLSS. The next column is the symbol column. Symbols can contain a maximum of 12 characters. The third column holds the comments. The length of the comment text depends on the actual length of the text in printouts. MiniLSS uses a proportional font, thus the number of characters in comment will vary, depending on the width of the letters used. The rightmost column is used to display the address state in the PLC.

When editing symbols or comment texts you can use all normal line editing commands of MiniLSS. (See the chapter *Line Editing Commands*.) Symbol or comment editing doesn't need to be ended by any special method; to edit the next symbol or comment just move the cursor to the desired field.

If undefined symbols are being used in the diagrams, they can be defined simply by entering the undefined symbol next to the desired address. This causes the temporary undefined address to be replaced with the defined one in all diagrams of the project. A faster way is to use the copy pointer (see explanation below). Undefined symbols will be discussed later.

In the Symbol Table Editor you can copy selected rows (both the symbol and the comment), the comment, or just one word. If the same comment is used many times it can first be copied to the copying buffer from which it can be fetched if necessary.

Similar or almost similar groups of points can be copied with the help of the copy pointer. When a copy command is given, the row that the pointer points to is copied into the current row, and the pointer moves to the next row. Note that if the current row has a defined symbol when the copy command is given, the symbol is not overwritten. The comment, however, will be overwritten.

When copying is done with a copy pointer, the text to be copied can be changed by replacing some part of the text with another text string. For example, if the following points are to be copied,

S1 Conveyor 1 position limit
M1 Conveyor 1 motor

it can be specified that the '1' string is to be replaced with the string '2'. Then the new points created by copying are

S2 Conveyor 2 position limit
M2 Conveyor 2 motor

All the '1' characters of the original text will be replaced with a '2' character.

Keyboard commands:

Ctrl **C** Copy word from previous row. The word that is above the cursor is copied to the cursor position.

Alt **P** Set the copy pointer to point to the current address.

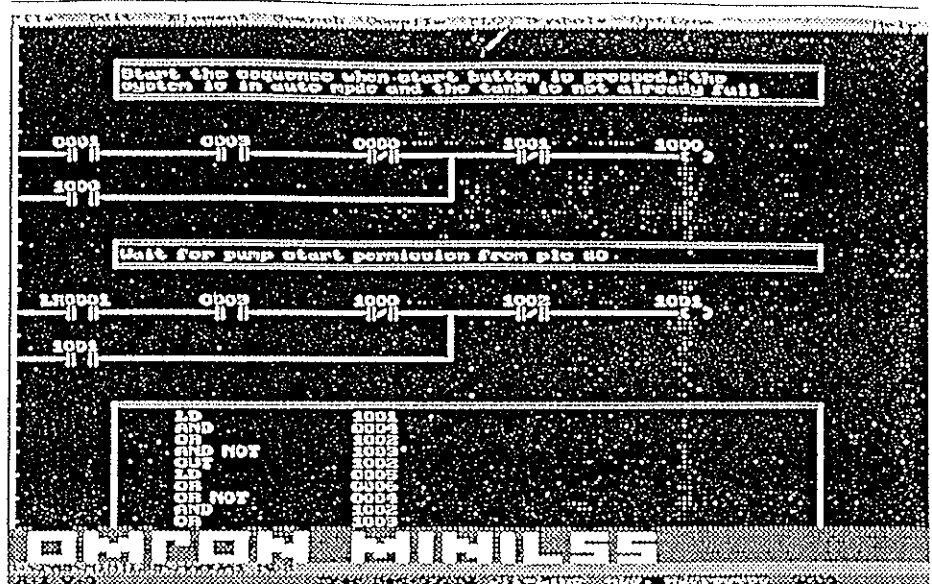
Ctrl **P** Copy the row pointed to by the copy pointer to the current row. (Also replace and copy.)

2.9.Editing Diagram Comments

Comments are free-formatted explanation texts in a diagram. They can be freely positioned in a diagram and are surrounded with a frame. The amount of rows inside the frame is unlimited.



A comment is created by moving the cursor to a desired location and selecting Comments from the Element menu. The comment may be edited by simply positioning the cursor inside the comment frame.


A typical editor screen with ladder diagram, instruction list and diagram comments appears as follows:





2.10. Line Editing Commands


When entering text, addresses, diagram comments, file names, editing the symbol table and so on, you can use all the line editing commands of MiniLSS. Line editing commands are all the same with one exception: The line-end command may vary. Typically when entering data with more than one row (as in the symbol table, or with a diagram comment), a field (or row) can be ended just by moving up or down. In one-row fields (e.g. when

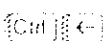
entering a contact address) the field must be ended with the  or  key.

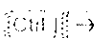
 Move cursor to the beginning of the field.


 Move cursor to the end of the field


 Move cursor one character left


 Move cursor one character right


 Move cursor one word left



 Move cursor one word right



 Move to the beginning of a comment field (in the Symbol Table Editor)



 Toggle insert/overwrite mode



 Delete the character under the cursor


 (Backspace) Delete the character to the left of the cursor.


  Delete the end of the word.

  Delete the end of field.

  Delete the beginning of the field.

  Delete the whole field.

 Accept the field and stop editing.

 Reject changes and stop editing. (Does not work with multiple-row fields.)

Reject changes (returns to the previous text) and continue editing.

In the above list the concept of field must be interpreted broadly, as it depends on the target of editing. It can be the whole line (symbol table, diagram comment) or a separate field (contact address, file name).

SECTION 3

Programming

3.1. Files

Everything that is entered, ladder diagrams instruction lists, diagram comments, symbols etc. must be saved to file before exiting MiniLSS. Each program is referred to by its name. The name can be a maximum of 8 characters long.

For one program MiniLSS will create several files which have different extensions. (DOS file names have an 8-letter name and a 3-letter extension). The following files will be created:

<name>.NDS Ladder diagrams, instruction lists and diagram comments will go to this file.

<name>.STB This file holds address symbols and comments. To retrieve your program from disk, .NDS and .STB files must both exist.

<name>.LOB This file contains the compiled program to be downloaded to the PLC. Also, uploading a program from the PLC produces this file.

<name>.MNE This file is created during compilation. Usually, you don't ever have to work with this file.

<name>.BND This is a backup file of previous .NDS file.

<name>.BST This is a backup file of previous .STB file.

When saving or opening a program file the file extension must be .NDS.

To restore the program from backup files, rename them (.BND to .NDS and .BST to .STB)

3.2. Designing the diagrams

Before starting a new project, select the correct PLC type and set the LR-Area size as needed. The LR Area settings affect the number of available DR words.

MiniLSS is intended to be used as much for the designing phase as for the programming phase. The exact positions of contacts and other elements are stored to the project file, and their positions on the screen will never change unless you intentionally move them. Syntax or logical integrity is not required until you decide to compile the program. This allows you to sketch incomplete diagrams, save the program and finish it at some other time. The use of *undefined address symbols* are provided to help in the designing phase.

3.3. Using symbols

Symbols are short declarations of address meanings. It is easier to remember that "1st scan" stands for the very first program scan after power-up than to remember that the address 0410 does the same.

MiniLSS accepts symbolic addresses, like "1st scan" as well as absolute addresses, like 0410. You may choose whether to use symbolic or absolute addressing, by selecting from the menu (*Symbols->Symbol Mode* or in LSS mode *PROGRAMMING->Symbol Mode*). The selected mode is seen on the status line at the bottom of the screen.

Note! In LSS mode the symbols are only displayed, all input must be absolute, in MiniLSS mode also the input can be symbolic.

If you select symbolic addresses, all addresses entered for contacts are symbolic. Thus if you enter text "0410" in symbol mode, it is treated as a symbol, **not** as address 0410. Symbol text is case sensitive: symbol 1s1 is not the same as symbol 1S1.

You can define the symbols in the Symbol Table Editor before using them in the Ladder Editor, or you can use them in ladder diagrams prior to defining them. If you use symbols that are not defined, you will get a message each time that you enter a symbol that has not yet been used. This will help you to detect typing errors at the moment you make them. If you mistype the symbol "1st scan" as "1stscan", you will get a message that a new symbol called "1stscan" has been entered.

Undefined symbols are assigned an address, called an *Undefined Address*, that begins with the letters UI followed by a unique sequence number. These UI addresses do not have any corresponding addresses in the PLC and they must be replaced with real PLC addresses before compiling.

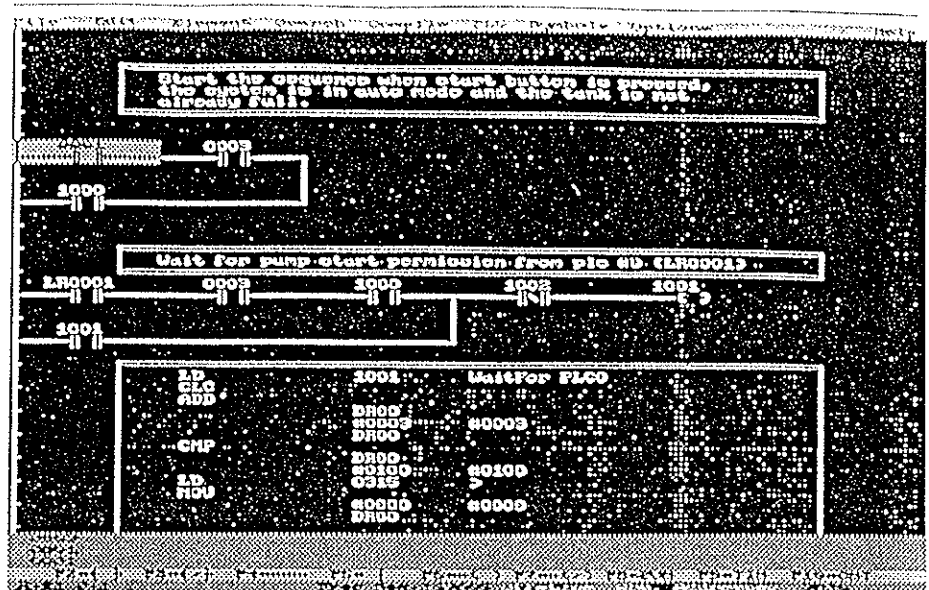
Using undefined symbols is useful when sketching diagrams. When changing diagrams, deleting portions of them or inserting other portions, you don't have to determine which real PLC addresses are needed. When you end up with something useful, you can see how many addresses are really needed and assign those to the real PLC addresses.

3.4. Using the clipboard

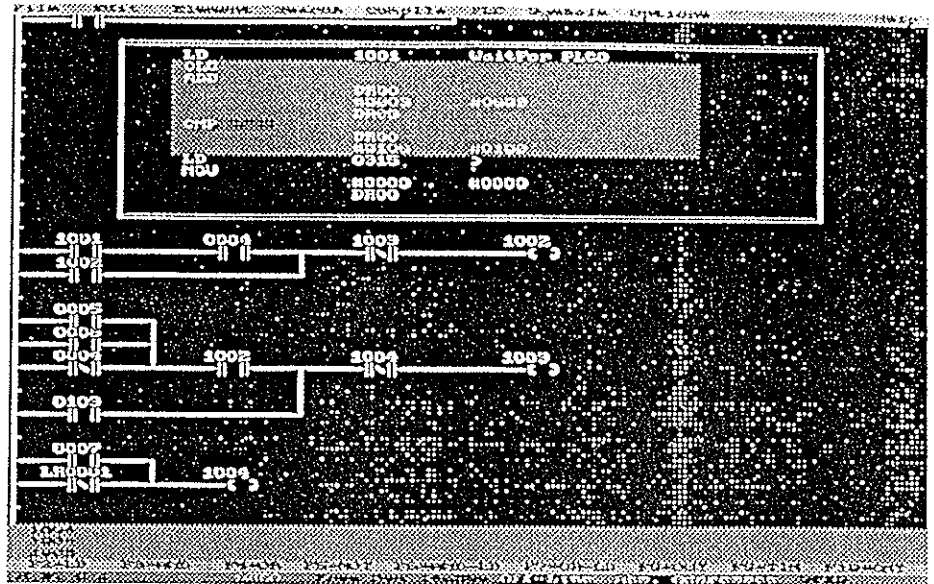
The clipboard is a temporary storage where you can store portions of diagrams or instruction lists. In one time, the clipboard holds only ladder elements or only instruction lines. Note that a whole instruction list element is a ladder element and can be copied together with other ladder elements, but a portion of an instruction list element (i.e. one or more instruction lines) is its own type and can be copied only inside an instruction list element or from one instruction list element into another.

3.4.1. Copying, cutting and pasting diagram portions

First select the portion to be stored by marking the area. To start marking, position the cursor at the top left corner of the area to be marked:

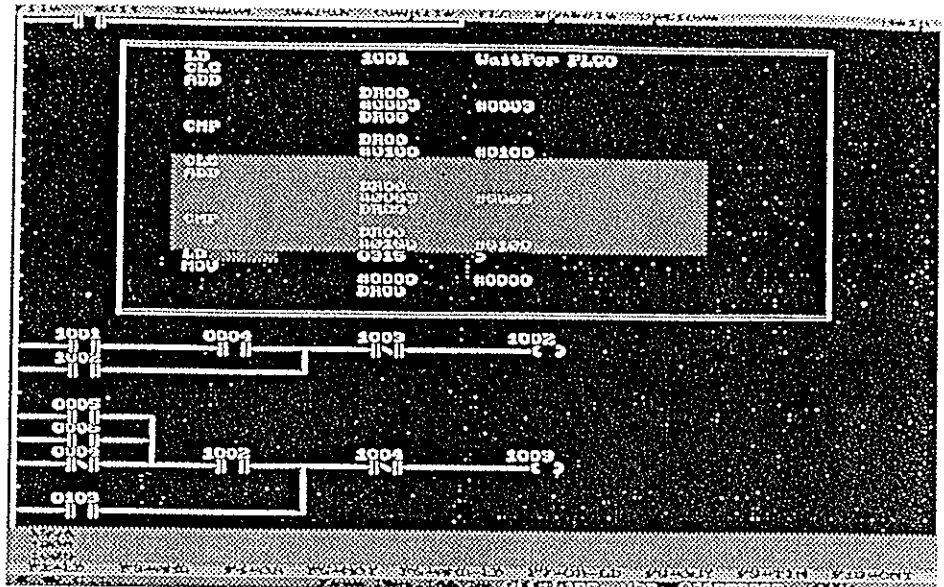


Before copying the area to be copied must be marked. To start marking, move the cursor to top of desired area and select *Mark* from the *Edit* menu (or in LSS mode *MARK* from the *PROGRAMMING* menu), or hold the shift key down and move the cursor to the last line to be included in the area.



SECTION 3

diagram area you must make room to insert data from the clipboard, but in an instruction list MiniLSS does it for you.



3.4.3. Clearing

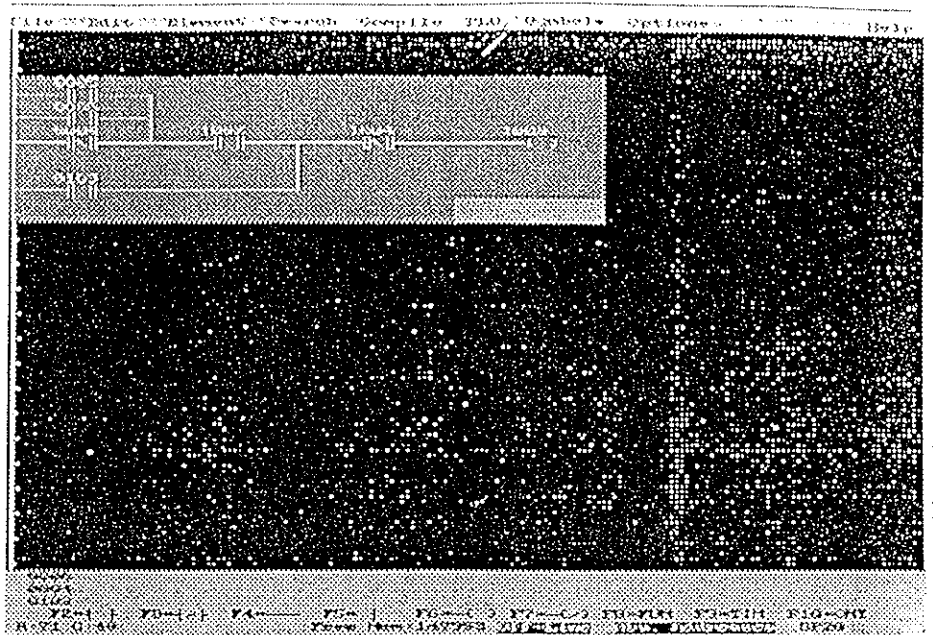
You can also clear the marked area by selecting *Clear* from the *Edit* menu (or in LSS mode *CLEAR* from the *PROGRAMMING* menu). This command clears the marked area and does not save it to the clipboard. Cleared data cannot be retrieved in any way. (Use the *Cut* command if you need to retrieve the data later from the clipboard.)

3.4.4. Converting program format with the clipboard

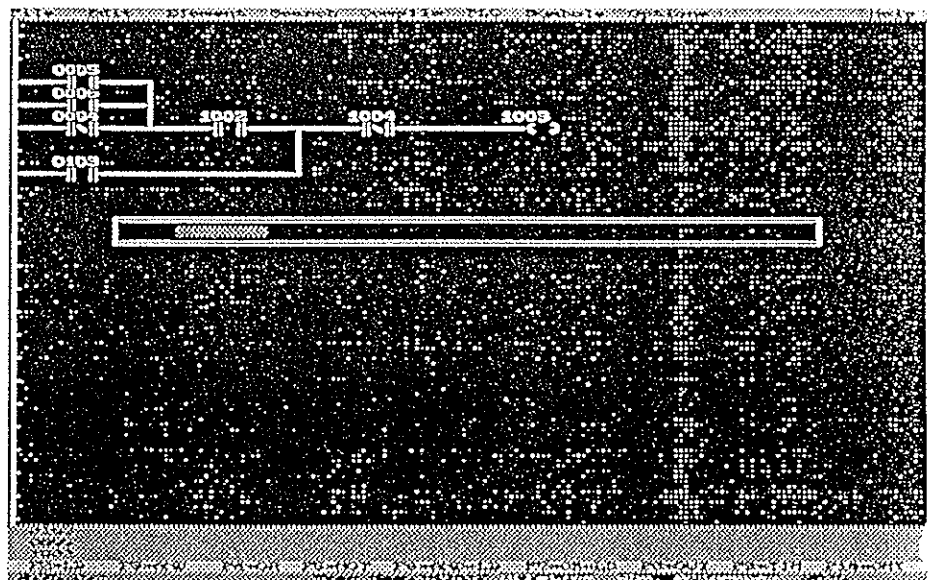
Note! Instruction list elements are not supported in LSS mode

The clipboard can receive data in two formats: ladder diagram format and instruction format. You can convert ladder diagrams to instructions by copying them onto the clipboard and by then pasting them into an instruction list element. If the data format is to be changed, direct pasting is not allowed. Instead you must select *Convert & Paste* from the *Edit* menu. Converting will succeed only if the diagrams are syntactically correct. Logical integrity is also required. Logical integrity means that all conditions (contacts) will end in an output/function/timer/counter

To convert from ladder diagram to instruction list, first mark the area

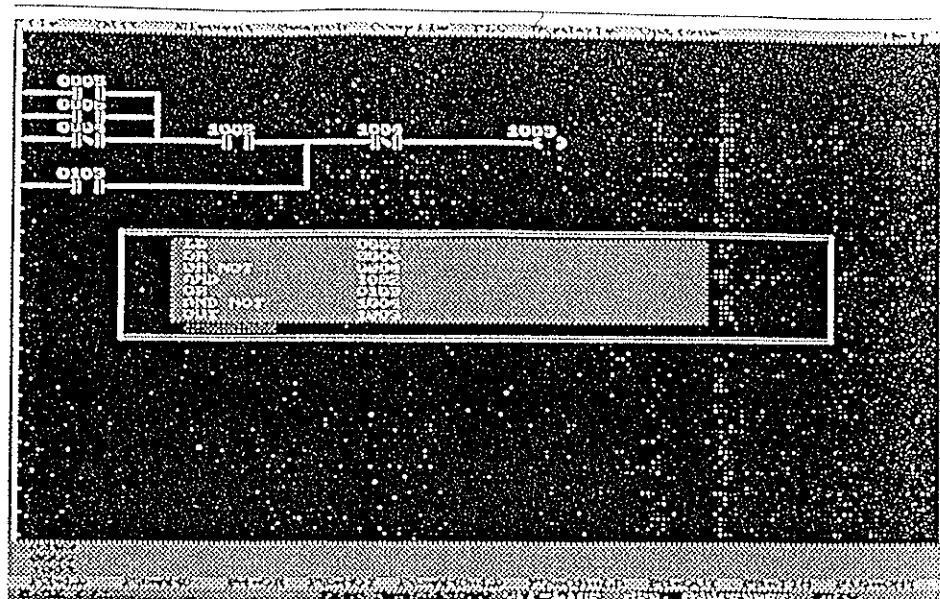


Insert an instruction list element or move the cursor inside an existing one:



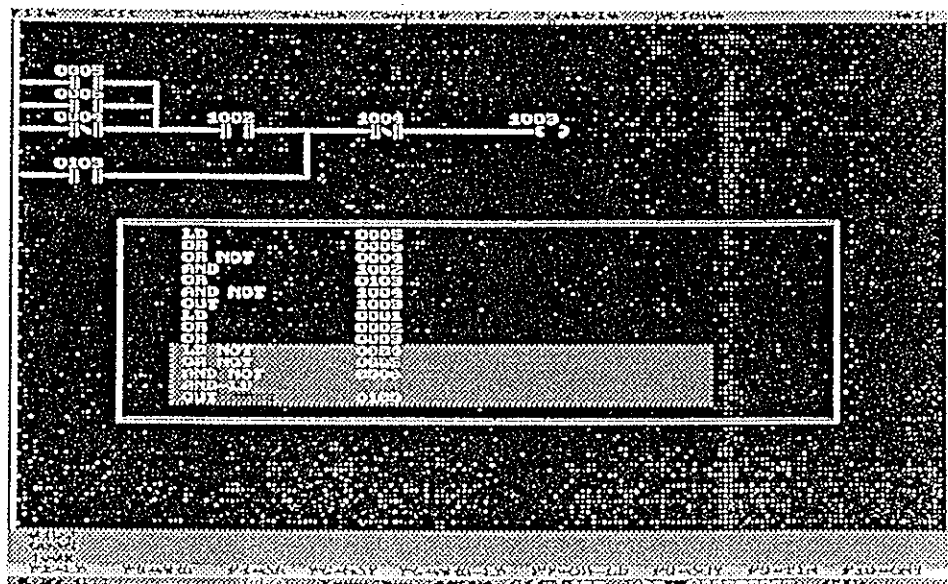
Select *Convert & Paste* from the *Edit* menu. If the marked area is syntactically correct, the corresponding instruction list block is inserted into the instruction list element.

SECTION 3

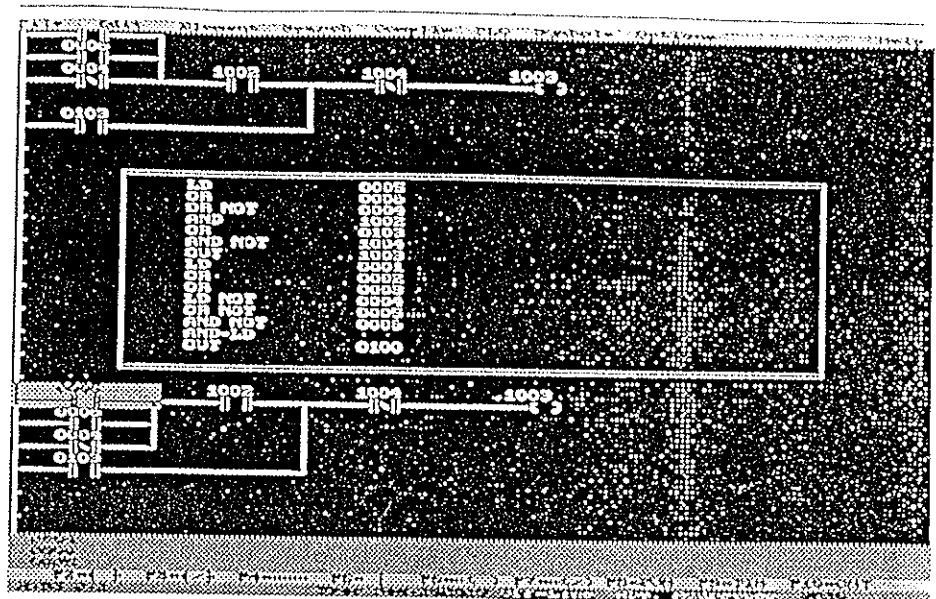


To convert from instruction list format to ladder format, first mark those instructions to be converted. Note that the instruction list must be logically whole: the last instruction must be an output-like instruction, each branch starting with an LD/LD-NOT instruction must end with outputs, etc.

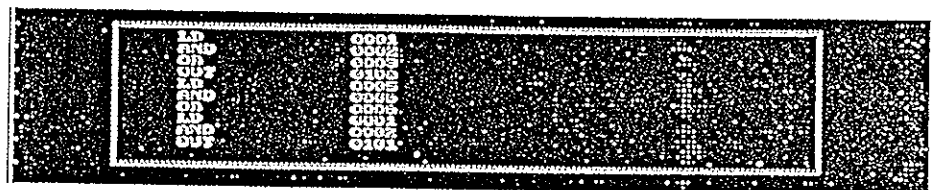
Move the cursor to the top of the desired area and select *Mark* from the *Edit* menu, or hold the shift key down and move the cursor to the last line to be included in the area:



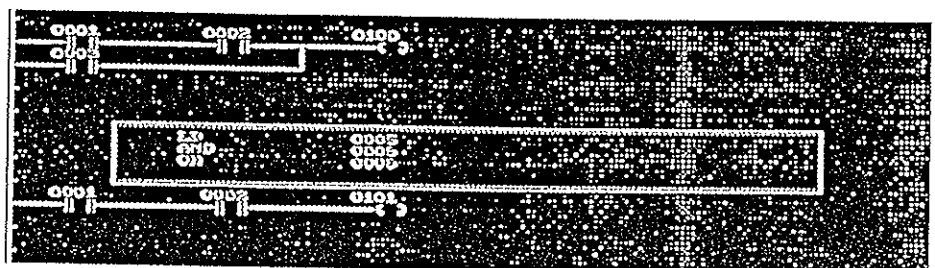
Select *Copy* from the *Edit* menu. Move the cursor in the diagram area to the top left corner of the area where the ladder elements are to be inserted, and select *Convert & Paste* from the *Edit* menu.



If the logical integrity of the instruction list copied onto the clipboard is not correct, those circuits found to be complete will be converted to ladder diagrams, and the rest of the instructions will be changed to new instruction list elements. For example, the following instruction list is not logically integral:



Converting it will produce partly diagrams and partly instruction lists:



The part converted to an instruction list has no effect on outputs and cannot be located to a diagram area

3.4.5. Saving the clipboard to a file

Once the data is copied into the clipboard, it can be saved to a disk file for a later use. You can collect useful program units to a "library," or you can merge different programs with the help of the clipboard.

Select the *Save Clipboard* item from the *File* menu (or in LCU mode *SAVE CLIPBOARD* from the *FILE MNGT* menu). A window will pop up asking for the file name. Enter the filename or select the filename from a pick list. If a list will appear if you specify a file name with valid characters.

3.4.6. Loading the clipboard from a file

The contents of the clipboard can be loaded from a disk file by selecting *Load Clipboard* from the *File* menu (or in LSS mode *SAVE CLIPBOARD* from the *FILE MNGT* menu). A window will pop up asking for the file name. Enter the filename or select the filename from a pick list. Pick list will appear if you specify a file name with wildcards, e.g. *.CBR.

Once the filename is given, the contents of the clipboard are loaded from the disk file. When the loading is complete, an Address Exchange Editor window will appear. Change the desired addresses and exit from the Address Exchange Editor (see the next chapter for using the Address Exchange Editor). Position the cursor and paste the data from clipboard as described earlier.

3.5. Using the Address Exchange Editor

The Address Exchange Editor can be used to change addresses of entire program, marked area or clipboard. To change addresses of the entire program, select *Change Addresses* from the *Edit* menu (in LSS mode *CHANGE ADDRESSES* from the *PROGRAMMING* menu), to change addresses to a marked area only, select *Change Area Addr* from the *Edit* menu (in LSS mode *CHANGE AREA ADDRESSES* in *PROGRAMMING* menu). Clipboard addresses can be changed just after the clipboard is loaded from a file. The Address Exchange Editor window is illustrated below:



Topline is the menu bar. In LSS mode the menu bar is absent and the menu can be called with **Ctrl M** or **End**

Original addresses, symbols and comments are shown in the left side of the window. In the right are two columns where new addresses/symbols can be defined. The leftmost of these columns is the address column (highlighted in the picture above), the rightmost is the symbol column. You can move the input focus between these columns with the tab key. Fill in the new desired address or the symbol to the highlighted field. The comment of the new address is shown in the window called "New address". In the picture below, one new address is entered:



Fill in all those addresses that you want to change. If the new address is left blank, the original address will not change. Thus, if the editing is terminated in a screen illustrated above, only the address DR0001 will be changed, and its new address will be DR0003. All other addresses will remain the same.

All menu selections are shown below:



Note! In LSS Mode the menu will look different. See chapter LSS Mode Menus for details.

MENUBAR SELECTIONS

<i>Exit</i>	Asks for a confirmation and if accepted changes addresses and closes the window
<i>Copy Addr</i>	Copy the original address to new address
<i>Addr by Symbol</i>	Find an address which has same symbol than the original one. Relevant only when handling a clipboard that is loaded from another program
<i>Copy&Replace</i>	Take the original address/symbol, replace part of it as specified by <i>Utils->Set Copy/Replace</i> and set it as the new symbol/address. If the cursor is in address column address string will be modified, if the cursor is in symbol column, symbol text will be modified
<i>Utils</i>	Opens Utils submenu
<i>Symbol Tab</i>	Lets you go to symbol table editor

UTILS SUBMENU SELECTIONS

<i>Copy Comment&Symbol</i>	Copy original symbol and comment to the symbol table, in position of the new address. Relevant only when handling a clipboard that is loaded from another program
--------------------------------	---

SECTION 3

Set Copy/Replace

With this command you can define a string which will be replaced by another string when copying.

SECTION 4 Printing Documents

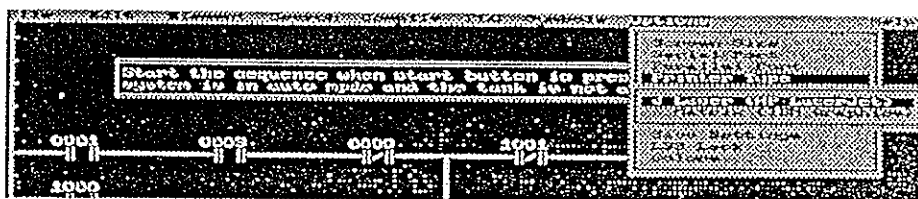
4.1. Choosing the printer

MiniLSS supports printers which can emulate those in the following list:

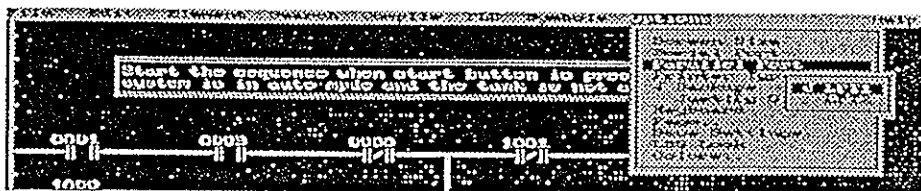
HP LaserJet+
HP LaserJet II
IBM Graphics

It is assumed that you are normally using a laser printer. However it is possible to make small printouts using a matrix printer. Check that your printer has one of the supported emulations and find out from the printer manual how that emulation is chosen.

The printer selection is made from the *Options* menu:



The parallel port where the printer is connected must also be set. That selection is available in the *Options* menu:



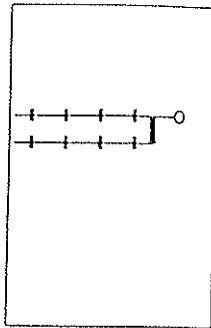
4.2. Orientation

Note: Orientation is only available for laser printers.

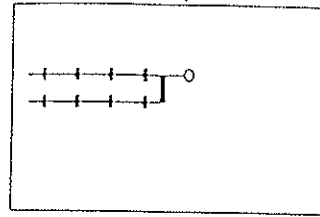
Page orientations can be selected as either portrait, landscape or mixed. Portrait orientation allows more lines to be drawn, and landscape orientation allows longer lines to be drawn.

SECTION 4

Portrait

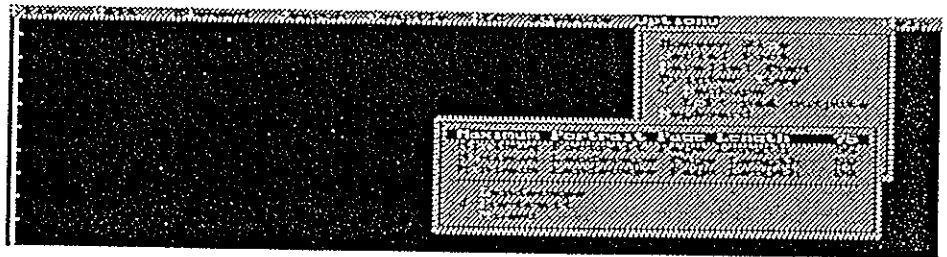


Landscape

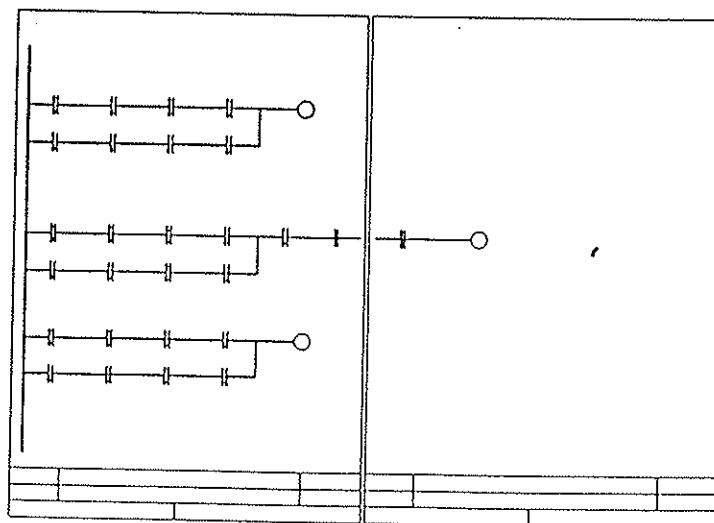


If you want only portrait pages, select *Portrait* on the (*Options->Page*) menu.

A page orientation can be set from the *Options* menu, with the *Page* selection. The *Page* menu appears as follows:

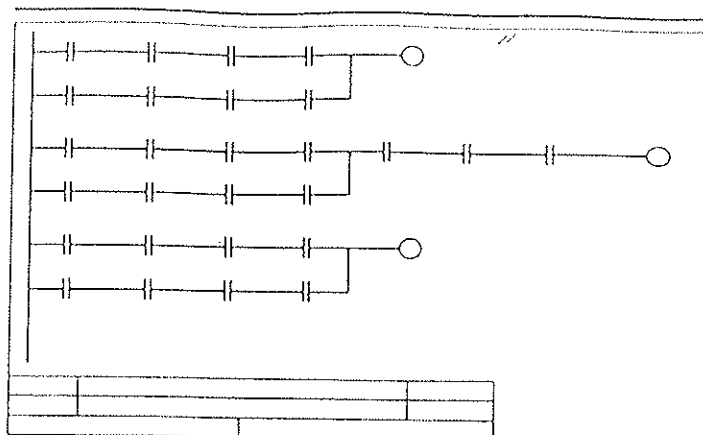


In portrait mode it is not guaranteed that the lines will fit on the page; i.e., you can draw longer lines with the editor than there is room on the paper. However, if long lines are detected, the page will be printed on two or three sheets, divided vertically:

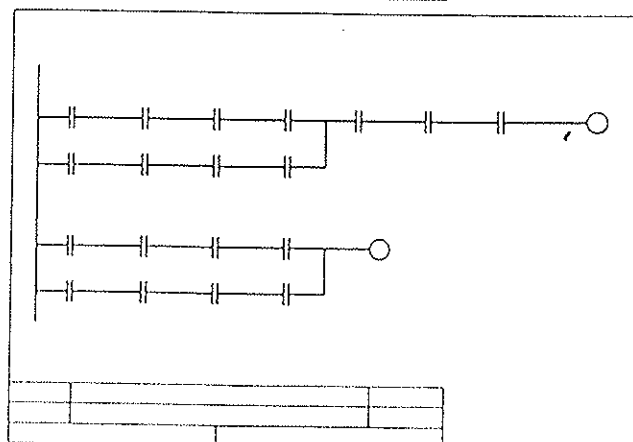
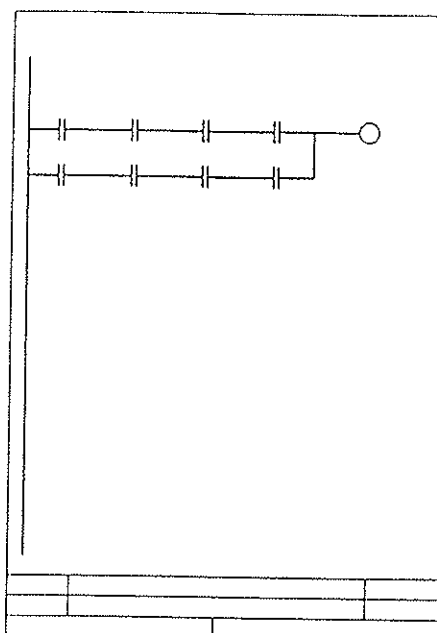


Both pages share the same page number.

If you select *Landscape* from the menu, there will only be landscape pages, no matter how wide or narrow the diagram is:



If *Mixed* is selected from the menu, there will be both portrait and landscape pages, depending on the width of diagrams. All diagrams that can be printed with portrait will be printed with a portrait orientation, the rest will be printed landscape:



4.3. Page Length

Page length can be set separately for portrait pages and landscape pages. There are two settings for both modes: Maximum Page Length and Minimum Page Length. The Maximum Page Length defines the maximum amount of print lines that are used for diagrams. The lines needed for a Header are not included in this length.

Minimum Page Length defines a page breaking zone. MiniLSS tries to make the page break on an empty line, as near to the defined maximum page length as possible. If there are no empty lines between Minimum Page Length and Maximum Page Length, the page will be cut on the line at Maximum Page Length. So, the page length will never be greater than the Maximum Page Length, and with one exception never smaller than the Minimum Page Length. The exception takes place in mixed print mode: switching from portrait to landscape may produce a portrait page that is shorter than the Minimum Page Length.

4.4. Headers

A header file is a separate freely-formatted text file, the contents of which are defined by the user. This is done with any text editor that produces a DOS text file. An editor is not supplied with MiniLSS.

In a header file you can use semi-graphic marks to draw lines and boxes. When editing a header be sure that you don't make it too large (too many lines). This may cause overprinting (output continuing onto the next page).

When printing the first page MiniLSS will search for a header file named <project>.HDR from your hard disk. If this file is not found, MiniLSS uses a standard header file delivered with MiniLSS. The name of the standard header file for laser printers is LASER.HDR and for matrix printers MATRIX.HDR. If the default file is not found, MiniLSS will display an error message and prompt for a new file name. If a selection is not made (you press **[Esc]**), no header is included in the listings.

4.4.1. Listing Codes in a Header File

In a header file references to a project to be listed or the time of listing can be made with text codes. The text code is replaced with the corresponding information when it is printed.

All the text codes are described in the following list:

- | | |
|-----|---|
| \$H | This code is replaced with an output type: for example, 'diagram', 'symbol table', etc. |
| \$P | Increasing page number. |
| \$F | Name of the project file. |
| \$D | Date. This is fetched from the PC's calendar. |

\$T Time. This is fetched from the PC's clock.

The following codes correspond to the data which is entered into the project information window (the project information window is accessed from the *Edit* menu) :

\$A Name of the designer.

\$C Project description.

\$R Version number.

Besides these predefined text codes, a user can define his own codes to be included in a header. A user code is included in the header by writing \$<code number>. The code function must be defined before it is used. What is more, the code must follow the syntax required by the emulation of the printer; e.g., HP LaserJet+.

The definition of a user code starts from the beginning of a line. The first character must be '{' (a brace) and then \$<code number><SPACE>. After <SPACE> is the control code. The structure of the user code is:

{ \$<code number><SPACE><control code>

In the control code for listing operations you will often use characters which can't be written with a text editor (for example <ESC>). This is solved by making it possible to write both normal characters and ASCII codes for characters in the control code. The syntax of code requires that texts with normal characters are separated with a single apostrophe ('); ASCII codes are written one at a time and preceded by a '#' mark; each field must be separated from the others by a comma (,).

For example:

{ \$3 #27,')s0S',#14,'Company Name',#15

For more information on printer control, refer to the corresponding Printer Manual.

4.4.2. Header Alignments

All listings are sent to the printer proportionally (laser printers). This means that the print head moves a distance relative to the width of a character; i.e., the 'i' character width is shorter than the 'x' character width. This may cause difficulties when aligning a header file; characters intended to be positioned below each other are not printed in the same position as they appear on the screen. With semi-graphic characters this problem does not exist; their width is always the same, and during printing there is a check that the printing position equals the position on the screen.

Alignment can be made with help of a marker (ASCII code #173). It is represented with an upside-down exclamation mark on the screen. A marker is printed as a <SPACE> character, and it moves the print head to a position that equals the position on the screen.

SECTION 4

Besides being used in headers, the marker may be used in diagram comments, where it can be inserted with the function key **F2**.

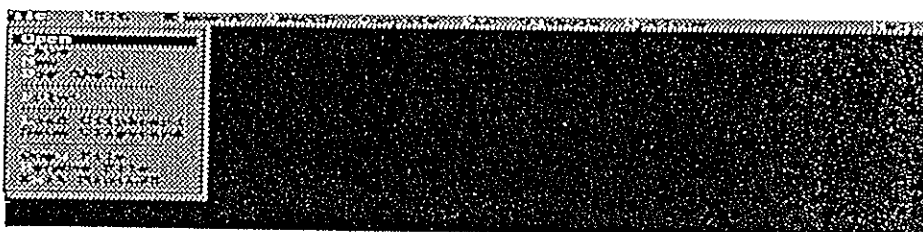
SECTION 5

Inside Menus

5.1.Ladder Editor Menus, MiniLSS Mode

5.1.1. Files

The *File* menu is used to load and save diagrams, to print diagrams and to exit the program.



Open

Loads diagrams from disk. After selecting this, a small window pops up asking for a filename. If you enter a filename with wildcards (e.g. *.NDS), a second window appears with a pick list where all matching filenames currently found on disk are displayed. To load diagrams, two files assigned to the filename must exist: <name>.NDS and <name>.STB. Note that diagrams (files with extension .NDS) and Symbol Tables (files with extension .STB) are always retrieved and saved together.

Save

Saves diagrams and Symbol Table to disk. After selecting this, a small window asking for a filename appears. The window may have a default filename already filled in. To accept the default name, just press **Enter**, otherwise fill in the required name and press **Enter**. As with the opening file procedure, if you enter a filename with wildcards, a pick list with matching filenames will be displayed. If you save a file with the same name as an existing file, the file originally on the disk will be overwritten. In this case you are asked for confirmation, to prevent accidentally overwriting other files.

New

Clears the working space and loads the default symbol table. Any changes made since the last save operation will be lost.

Dos Shell

Lets you go to DOS, leaving the program and diagrams in memory. To return back from DOS type EXIT at the DOS prompt. When returning, make sure that you are in the same directory.

Print

Opens the *Print* submenu

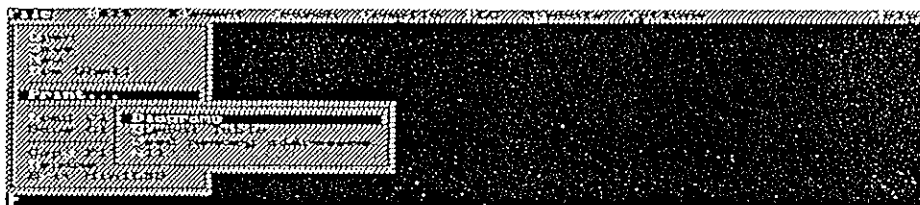
Load Clipboard

Loads the clipboard contents from a disk file, previously created with *Save Clipboard* command. After selecting this, a window asking for filename appears. As with the other file procedures, if you enter a filename with wildcards, a pick list with matching filenames will be displayed. Use Save/Load Clipboard when merging programs.

SECTION 5

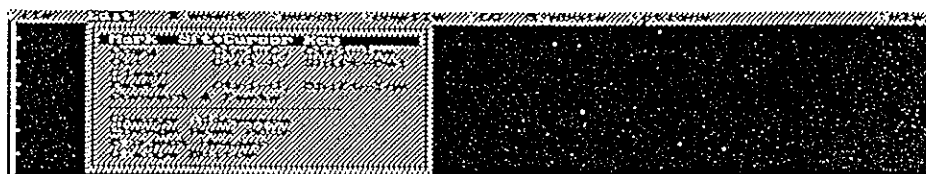
<i>Save Clipboard</i>	Saves the clipboard contents to a disk file. After selecting this, a window asking for filename appears. As with the other file procedures, if you enter a filename with wildcards, a pick list with matching filenames will be displayed. Use Save/Load Clipboard when merging programs.
<i>Copy File</i>	Lets you copy files, one at a time.
<i>Rename File</i>	Lets you rename files, one at a time.
<i>Exit</i>	Terminates the program and returns back to DOS. If you have made any changes since last file save, they will be lost. In this case you are asked to confirm exiting.

The *Print* menu lets you select the items to be printed.



<i>Diagrams</i>	Prints all diagrams and instruction lists.
<i>Symbol Table</i>	Prints the Symbol Table. Only those addresses will be printed that have symbols or comments attached.
<i>Used Memory Areas</i>	Print those addresses that are directly used in the program.
<i>All</i>	Print diagrams, symbol table and used memory areas.

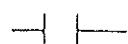
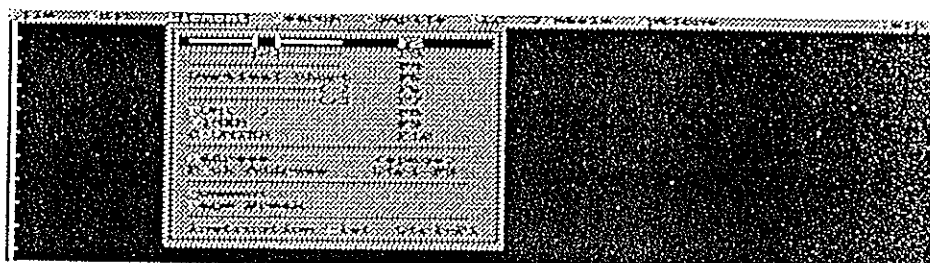
5.1.2. The Edit-Menu



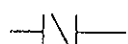
<i>Mark</i>	Marks the area to be copied/moved to the paste buffer. Marking can also be done by pressing the shift + cursor key with a normal PC keyboard.
<i>Copy</i>	Copies the marked area to a paste buffer.
<i>Cut</i>	Moves the marked area to a paste buffer and removes it from its original location.
<i>Clear</i>	Clears the contents of marked area, does not affect the paste buffer. Note: there is no way to get the cleared data back.
<i>Paste</i>	Pastes the data from the paste buffer to the cursor location.
<i>Convert & Paste</i>	The paste buffer can contain ladder diagrams or instruction lines. Use this command to convert instruction lines to ladder diagrams or vice versa.

- Change Addresses* Open the Address Exchange Editor with all the addresses found in the program
- Change Area Add* Open the Address Exchange Editor with all the addresses found in the marked area
- Project Header* Sets general information about the project: Author, Project Name and Version ID. These texts will appear in the listing header.
- Change Editor* Toggles between the full-screen instruction editor and the diagram editor.

5.1.3. Diagram Elements



Draws an open contact to the cursor position.



Draws a closed contact to the cursor position.



Draws a horizontal short to the cursor position.

Vertical short

Draws vertical short to the cursor position.



Draws an output to the cursor position.



Draws an inverted output to the cursor position.

FUN()

Lets you specify the function and draws it to the cursor position.

TIMER

Draws a timer to the cursor position.

COUNTER

Draws a counter to the cursor position.

Address

Lets you specify the address of a contact/output.

Pick Address

Picks the current address from the Symbol Table and assigns it to the contact underneath the cursor. The current address is the one where cursor was at your last exit from the Symbol Table Editor.

Comment

Inserts a diagram comment at the cursor position (Shortcut key for comment is "C")

Page Break

Inserts a page break at the cursor position

Instruction List

Inserts an instruction list element at the cursor position

5.1.4. Searching



Search Bit Address Lets you enter a bit address and searches for its first occurrence starting from the cursor location. All contacts, outputs, functions and instruction lists are searched.

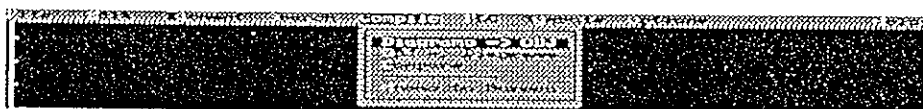
Search Word Address Lets you enter a word address and searches for its first occurrence starting from the cursor location. All contacts, outputs, functions and instruction lists are searched.

Search Instr Lets you specify the instruction and optionally, its operands. If an operand/address is left blank, any operand/address fulfills the criteria.

Search Next Searches for the next occurrence of an address/instruction previously given with the *Search Bit Address*, *Search Word Address* or *Search Instruction* selections. (Shortcut key for Search Next is "N")

Goto Line Positions the cursor at the line specified.

5.1.5. Compiling



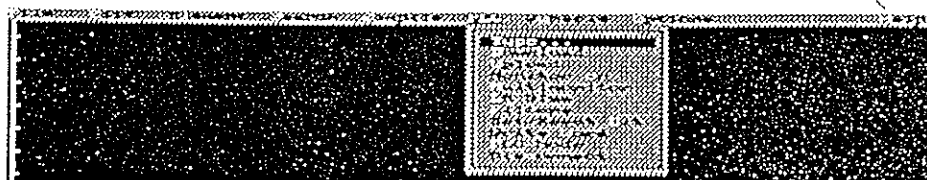
Diagrams -> Obj Compiles diagrams to a PLC object code file. The object file will have an .LOB extension and can be transferred to the PLC.

Obj -> Diagrams Compiles an object file to diagrams. The object file is created by uploading the contents of the PLC memory in the PLC menu.

Compare Compares currently loaded diagrams with an object code file.

Mnemo To Screen Compiles diagrams to the screen window. All syntax checks will be performed.

5.1.6. PLC Utilities



Type Opens the PLC selection menu.

Download Downloads an .LOB file to the PLC.

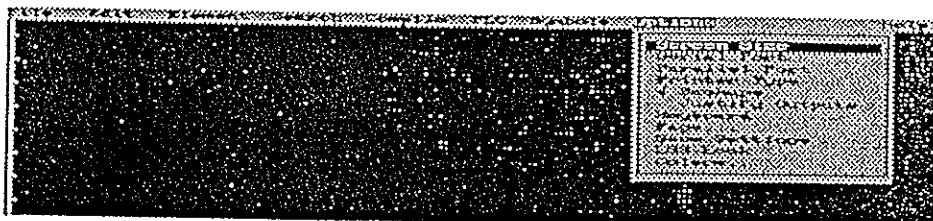
<i>Upload</i>	Uploads contents of PLC program memory.
<i>Monitor On</i>	When monitor mode is on, the statuses of PLC addresses are shown in diagrams. (Shortcut key: "M")
<i>Multimonitor</i>	Opens a multimonitor page where statuses of several PLC addresses are shown.
<i>LR Size</i>	Sets the LR size.
<i>Filters</i>	Sets the values of input filters.
<i>Set/Reset Bit</i>	Forces specified bit ON/OFF. (Shortcut key: "B")
<i>Write Word</i>	Changes the value of a word address in the PLC. (Shortcut key: "W")
<i>PLC Mode</i>	Lets you set the PLC to RUN mode and to PROGRAM mode
<i>Disconnect</i>	Once you have established a connection to a PLC, you can not change the PLC number unless loading another program or selecting <i>Disconnect</i> .

5.1.7. Symbols



<i>Symbol Table</i>	Go to the Symbol Table Editor.
<i>Symbol Mode</i>	When the symbol mode is on, the symbols (or names) of addresses are displayed instead of numerical addresses. (Shortcut key: "M")

5.1.8. Options



<i>Screen Size</i>	Lets you select the screen height: 25 lines or 43/50 lines (43/50 is preferable if your hardware supports it).
<i>Serial Port</i>	Lets you select the serial port to be used: COM1 or COM2.
<i>Parallel Port</i>	Lets you select the port where the printer is connected: LPT1/LPT2.
<i>Printer Type</i>	Lets you select the printer type.

SECTION 5

Backups

When this option is on, the old files will be saved when storing diagrams. Extensions for backup files are .BND for the diagram file and .BST for the symbol table file.

Justify Outputs

Display the outputs, timers, counters and functions justified to right edge of the screen.

Keyboard

Lets you select the typematic delay and the typematic rate of your keyboard.

Page

Lets you select the length of the paper used as well as its orientation.

Save Settings

Saves the current settings to a disk file in the current directory.

LSS-Look

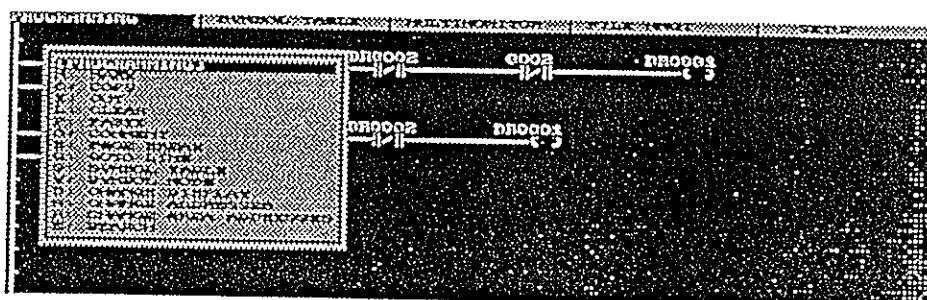
Change the program outlook to LSS-style

Colors

Specify the screen colors

5.2. Ladder Editor Menus, LSS Mode

5.2.1. Programming

*Mark*

Marks the area to be copied/moved to the paste buffer. Marking can also be done by pressing the shift + cursor key with a normal PC keyboard.

Copy

Copies the marked area to a paste buffer.

Cut

Moves the marked area to a paste buffer and removes it from its original location.

Clear

Clears the contents of marked area, does not affect the paste buffer. Note: there is no way to get the cleared data back.

Paste

Pastes the data from the paste buffer to the cursor location.

Comment

Inserts a diagram comment at the cursor position.

Page Break

Inserts a page break at the cursor position

Goto Line

Positions the cursor at the line specified.

Symbol Table

Go to the Symbol Table Editor.

Symbol Mode

When the symbol mode is on, the symbols (or names) of addresses are displayed instead of numerical addresses.

Change Display

Toggles between the full-screen instruction editor and the diagram editor.

Change Addresses Open the Address Exchange Editor with all the addresses found in the program

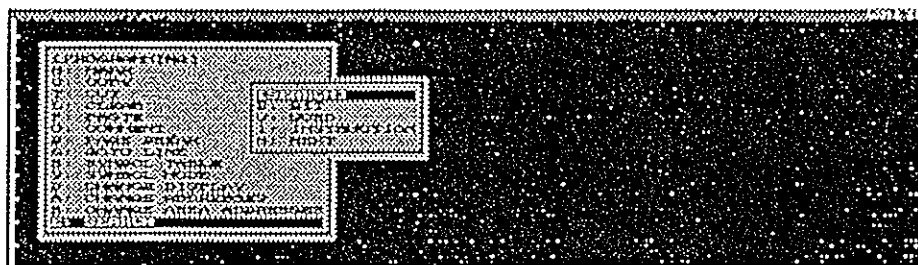
Change Area Addresses

Open the Address Exchange Editor with all the addresses found in the marked area

Search

Opens the Search submenu

5.2.2. Search



Bit Lets you enter a bit address and searches for its first occurrence starting from the cursor location. All contacts, outputs, functions and instruction lists are searched.

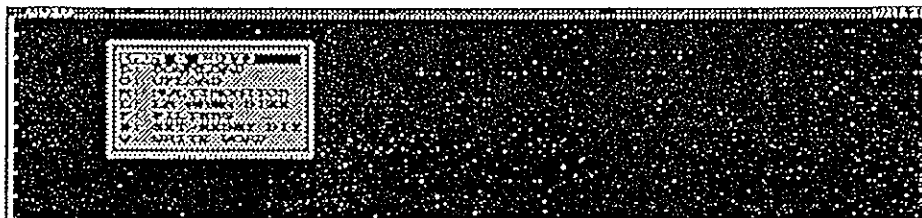
Word Lets you enter a word address and searches for its first occurrence starting from the cursor location. All contacts, outputs, functions and instruction lists are searched.

Instruction Lets you specify the instruction and optionally, its operands. If an operand/address is left blank, any operand/address fulfills the criteria.

Next Searches for the next occurrence of an address or instruction previously given with the *Search Bit*, *Search Word* or *Search Instruction* selections.

5.2.3. When PLC Connection is ON

The PROGRAMMING menu will change when you set the PLC connection on. The menu will appear as following:



Download Downloads an .LOB file to the PLC.

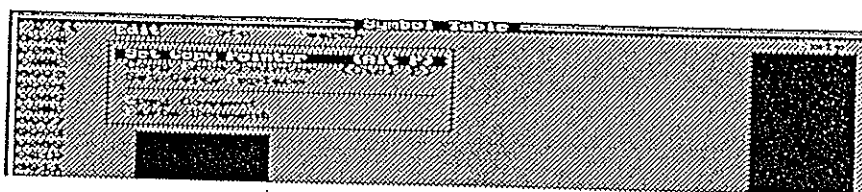
Upload Uploads contents of PLC program memory.

Multimonitor Opens a multimonitor page where statuses of several PLC addresses are shown.

<i>LR Size</i>	Sets the LR size.
<i>Filters</i>	Sets the values of input filters.
<i>Set/Reset Bit</i>	Forces specified bit ON/OFF.
<i>Write Word</i>	Changes the value of a word address in the PLC.

5.3. Symbol Table Editor menus, MiniLSS Mode

5.3.1. Edit Menu



Set Copy Pointer Sets the copy pointer to the current line. A small arrow left in the symbol column marks the position of the copy pointer.

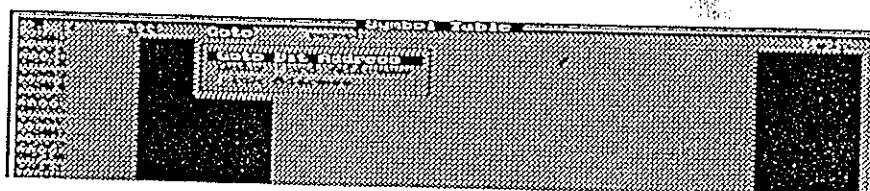
Copy From Pointer Copies from the line specified by the copy pointer to the current line and moves the copy pointer one line downwards

Set Copy/Replace With this command you can define a string which will be replaced by another string when copying.

Copy Comment Copies comment text to the paste buffer

Paste Comment Copies comment text from the paste buffer to the current line.

5.3.2. Goto Menu

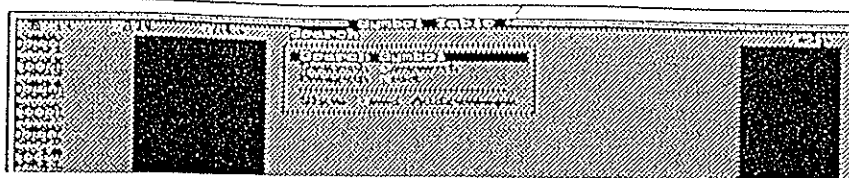


Goto Bit Address Moves the current line to the bit address specified.

Goto Word Address Moves the current line to the word address specified.

Pick Address Opens a pick window, where recently searched addresses can be selected.

5.3.3. Search Menu



Search Symbol Search for an occurrence of text in symbol texts.

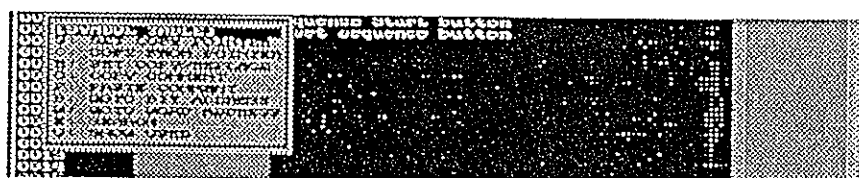
Search Comment Search for an occurrence of text in comment texts

Search Next Repeat the last search.

Show Used Addresses

Create a cross reference and indicate addresses that are directly used in the program.

5.4. Symbol Table Editor menus, LSS Mode



Set Copy Pointer Sets the copy pointer to the current line. A small arrow left in the symbol column marks the position of the copy pointer.

Copy From Pointer Copies from the line specified by the copy pointer to the current line and moves the copy pointer one line downwards

Set Copy/Replace With this command you can define a string which will be replaced by another string when copying.

Copy Comment Copies comment text to the paste buffer

Paste Comment Copies comment text from the paste buffer to the current line.

Goto Bit Address Moves the current line to the bit address specified.

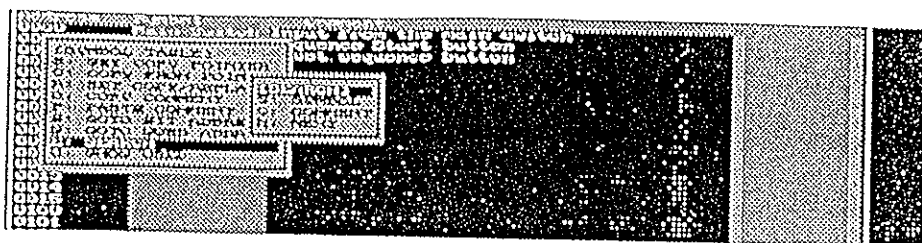
Goto Word Address

Moves the current line to the word address specified.

Search Opens the Search submenu

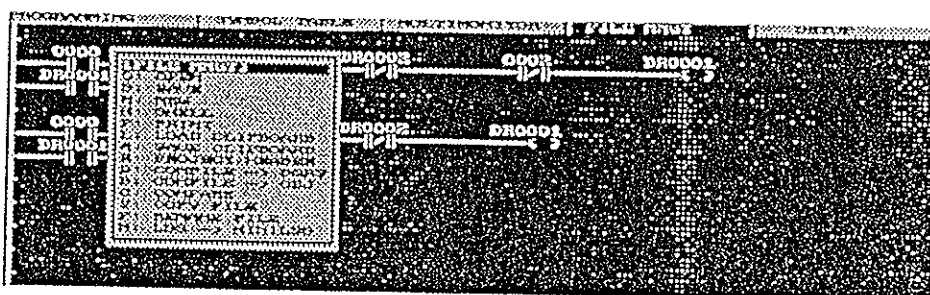
Show Used Create a cross reference and indicate which addresses are directly used in the program.

5.4.1. Search submenu



<i>Symbol</i>	Search for an occurrence of text in symbol texts.
<i>Comment</i>	Search for an occurrence of text in comment texts
<i>Next</i>	Repeat the last search.

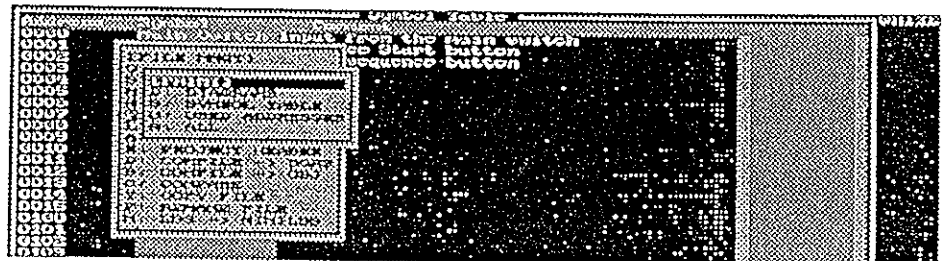
5.5. File management menu, LSS Mode



<i>Open</i>	Loads diagrams from disk. After selecting this, a small window pops up asking for a filename. If you enter a filename with wildcards (e.g. *.NDS), a second window appears with a pick list where all matching filenames currently found on disk are displayed. To load diagrams, two files assigned to the filename must exist: <name>.NDS and <name>.STB. Note that diagrams (files with extension .NDS) and Symbol Tables (files with extension .STB) are always retrieved and saved together.
<i>Save</i>	Saves diagrams and Symbol Table to disk. After selecting this, a small window asking for a filename appears. The window may have a default filename already filled in. To accept the default name, just press <input type="button" value="Enter"/> , otherwise fill in the required name and press <input type="button" value="Enter"/> . As with the opening file procedure, if you enter a filename with wildcards, a pick list with matching filenames will be displayed. If you save a file with the same name as an existing file, the file originally on the disk will be overwritten. In this case you are asked for confirmation, to prevent accidentally overwriting other files.
<i>New</i>	Clears the working space and loads the default symbol table. Any changes made since the last save operation will be lost.
<i>Shell</i>	Lets you go to DOS, leaving the program and diagrams in memory. To return back from DOS type EXIT at the DOS prompt.
<i>Print</i>	Opens the <i>Print</i> submenu.

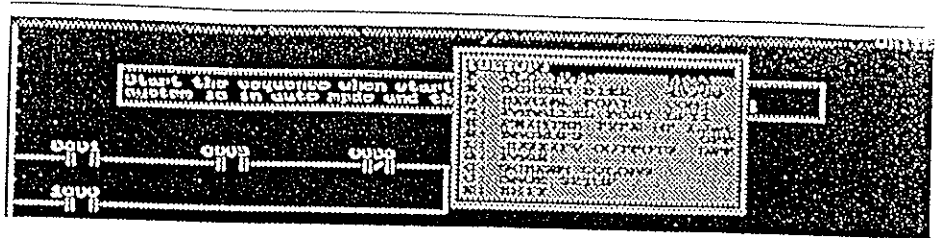
- Load Clipboard* Loads the clipboard contents from a disk file, previously created with *Save Clipboard* command. After selecting this a window asking for filename appears. As with the other file procedures, if you enter a filename with wildcards, a pick list with matching filenames will be displayed. Use *Save/Load Clipboard* when merging programs.
- Save Clipboard* Saves the clipboard contents to a diskfile. After selecting this a window asking for filename appears. As with the other file procedures, if you enter a filename with wildcards, a pick list with matching filenames will be displayed. Use *Save/Load Clipboard* when merging programs.
- Project Header* Sets general information about the project: Author, Project Name and Version ID. These texts will appear in the listing header.
- Compile -> Dgrs* Compiles an object file to diagrams. The object file is created by uploading the contents of the PLC memory in the *MON&EDIT* menu.
- Compile -> Obj* Compiles diagrams to a PLC object code file. The object file will have an .LOB extension and can be transferred to the PLC.
- Compare* Compares currently loaded diagrams with an object code file.
- Copy File* Lets you copy files, one at a time.
- Rename File* Lets you rename files, one at a time.
- LSS -> MINILSS* Changes the program outlook to MiniLSS-style.

Print submenu has the following selections:



- Diagrams* Prints all diagrams and instruction lists.
- Symbol Table* Prints the Symbol Table. Only those addresses will be printed that have symbols or comments attached.
- Used Addresses* Print those addresses that are directly used in the program.
- All* Print diagrams, symbol table and used memory areas.

5.6. Setup menu, LSS Mode



<i>PLC Type</i>	Opens the PLC selection menu.
<i>Screen Size</i>	Lets you select the screen height: 25 lines or 43/50 lines (43/50 is preferable if your hardware supports it).
<i>Serial Port</i>	Lets you select the serial port to be used: COM1 or COM2.
<i>Parallel Port</i>	Lets you select the port where the printer is connected: LPT1/LPT2.
<i>Printer Type</i>	Lets you select the printer type.
<i>Backups</i>	When this option is on, the old files will be saved when storing diagrams. Extensions for backup files are .BND for the diagram file and .BST for the symbol table file.
<i>Justify Outputs</i>	Display the outputs, timers, counters and functions justified to right edge of the screen.
<i>Page</i>	Lets you select the length of the paper used and its orientation.
<i>Screen Colors</i>	Lets you edit the screen colors used.
<i>Save Setup</i>	Saves the current settings to a disk file in the current directory.
<i>Exit</i>	Terminates the program and returns back to DOS. If you have made any changes since last file save, they will be lost. In this case you are asked to confirm exiting.

Sometimes it is necessary to observe several PLC addresses simultaneously, although they are contained in separate diagrams. This can be done with the help of Multimonitoring Pages. You can collect one screenful of points (bits and words) into these pages from various memory areas of the PLC. These pages can also be saved to a file from which they can be retrieved if needed.

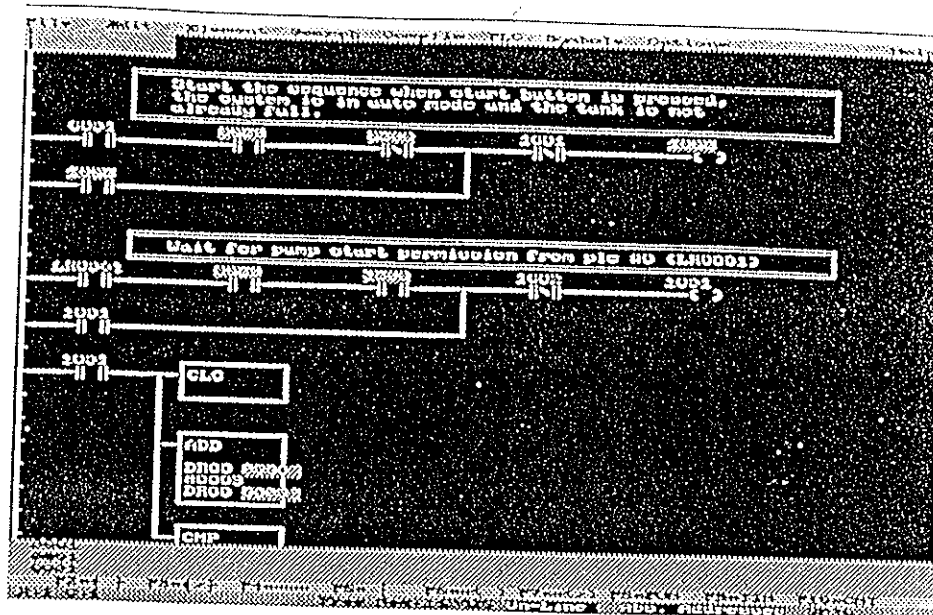
6.1. Monitoring a Diagram

In LSS-Mode, the PLC connection is made by pressing **Ctrl** **O**. The function key guide changes to following:



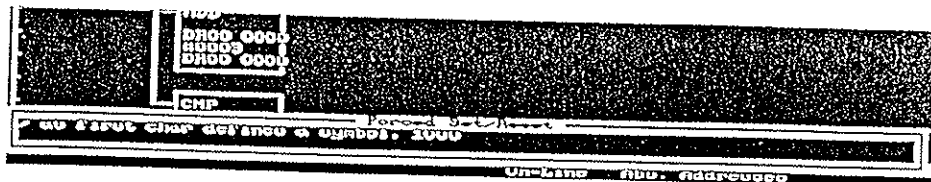
Press **F1** to establish connection to PLC.

The 1-state of a PLC's bit address is represented with a brighter color than the 0-state of the address (with default color settings). Word values are displayed to the right of operands:

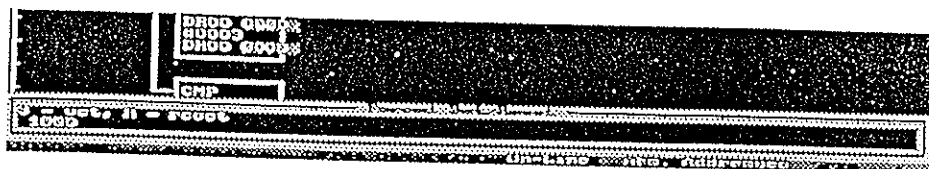


The contents of PLC addresses can be changed in Monitoring mode with the selections *Set/Reset Bit* and *Write Word* in the *PLC* menu. In LSS Mode the menus are different. See end of this chapter for LSS mode menus.

After selecting *Set/Reset Bit* a window opens at the bottom of the screen asking for the bit address. If the cursor is over a contact when selecting *Set/Reset Bit*, the default address is the address of that contact; otherwise the default address is the address that was last set:



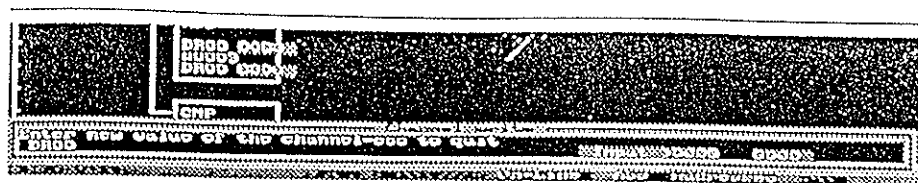
Fill in the required address, or to accept a default address, press **Enter**. The address can also be entered as symbolic by adding a 'V' in front of the symbol. The window changes, now displaying the address and its current state:




By pressing **[S]** the bit is forced on and by pressing **[R]** the bit is forced off

Note! When changing values, the new value is only written once into the PLC. There is nothing to prevent the PLC from automatically changing this value.

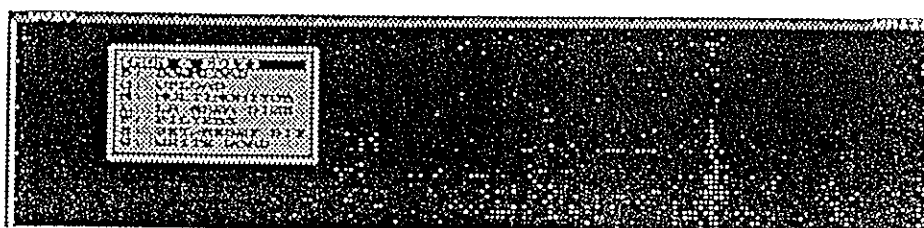
Note! Bit values are written word based: contents of the word are read, the bit in question is changed and the whole word is written back to the PLC. If PLC program has changed other bits in that word in between read and write operation, states of those bits are restored back to their states at read time.



Writing word values is begun with the *Write Word* selection in the *PLC* menu. After this selection a window appears at the bottom of the screen, asking for the word address. After entering the address, the window changes and displays the word address and its current value. Enter the desired value into the input field and press . The new value is written to the PLC.

Note! When changing values, the new value is only written once into the PLC. There is nothing to prevent the PLC from automatically changing this value.

In LSS Mode the PROGRAMMING menu changes after the PLC Connection is established. The menu will look like the following:



6.2. Monitoring the Symbol Table

The Symbol Table can be scanned in Online mode where the value of every PLC address is displayed in the rightmost field of the Symbol Table. However, MiniLSS cannot be set to Online mode from the Symbol Table - this must be done from the Diagram Editor.

Bit addresses are represented with four "*" marks, which grow brighter in the bit's 1-state, and grow dimmer in the 0-state. The values of channel addresses are represented by numeric strings.

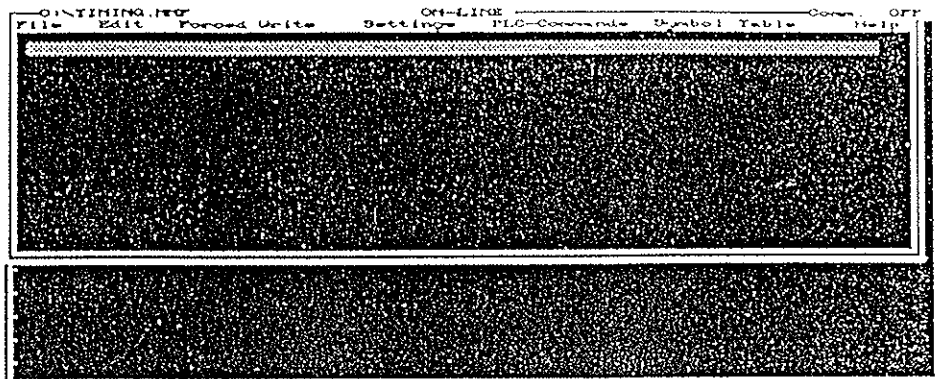
6.3. Multimonitoring Pages

A Multimonitoring Page is accessed from the Diagram Editor with the *Multimonitor* selection in the *PLC* menu. If a connection has already been established, MiniLSS will automatically switch to Online mode, and when returning to the Diagram Editor the previous Online/Offline mode will be

reinstated. If there is no connection, Multimonitoring Pages may be scanned and edited in Offline mode.

6.3.1. Structure of a Multimonitoring Page

A Multimonitoring Page is illustrated below.



A Multimonitoring Page reserves the upper half of the screen if the screen is set to 43/50 line mode, or the whole screen if it is set to 25 line mode.

The name of the Multimonitoring Page definition file is displayed in the upper left-hand corner. The upper right-hand corner notifies whether a connection to the PLC exists or not. If the connection is lost for some reason, the states of the displayed points will not necessarily indicate the actual states of PLC addresses.

The next row is the menu bar. The menu bar can be accessed with the **[Esc]** key or by holding the **[Alt]** key down and pressing any of the highlighted characters.

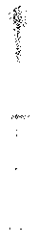
In LSS Mode the menu bar is absent, and the menu can be called with **[Ctrl] [M]** or **[End]**.

In LSS Mode when the menu is displayed, the screen appears as follows:



Note that even though the menu selections described in this chapter are based on MiniLSS mode, the same commands can be achieved also in LSS menu when LSS mode is selected.

The rest of the area is reserved for defining addresses to be monitored. Addresses, symbols and comments are displayed here. A filled Multimonitoring Page appears as follows:

 $\frac{1}{2}$ 

2

6.3.3. Changing PLC states in the Multimonitoring Page

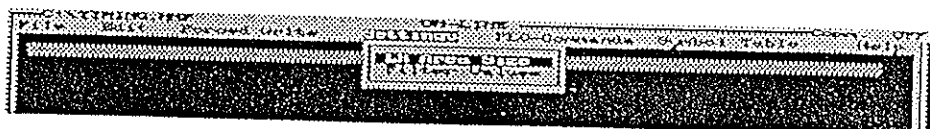
The contents of the PLC memory can be changed with selections in the *Forced Write* menu. The menu appears as follows:



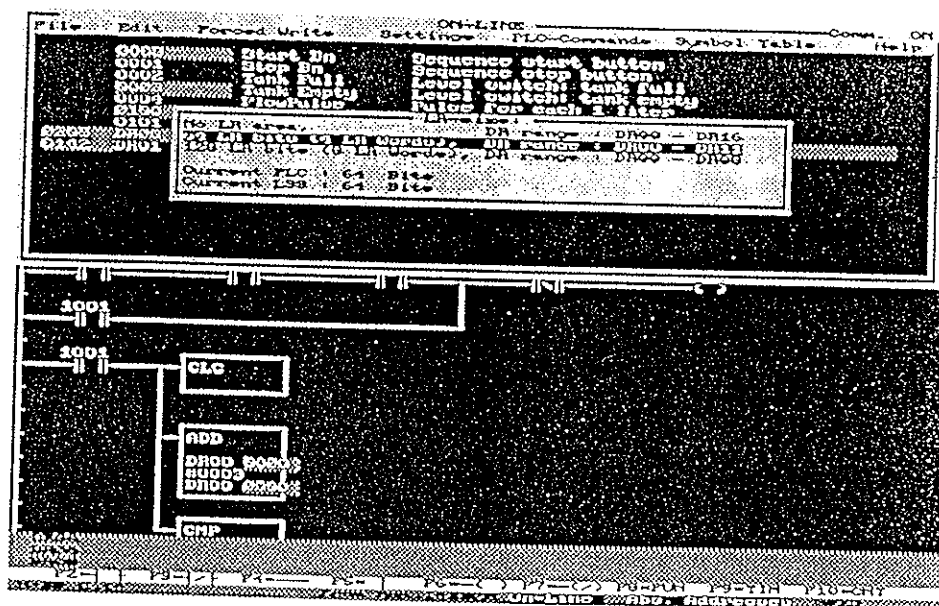
The procedure for *Set/Reset* and *Write Word* commands are similar to those in the Diagram Editor. See chapter *Monitoring a Diagram*.

6.3.4. Setting LR Area Size




The LR area size can be set in the Multimonitoring Page, or directly from the Diagram Editor. Select *LR Area Size* from the *Settings* menu:



A window with current LR area settings will appear. The LR area size, as well as the remaining DR area size is displayed in the window.



The possible selections for LR area size are 0 bits, 64 bits, or 128 bits. The selection will also affect the number of available DR words, as shown in the window. At the bottom of the window are the LR area sizes currently found in the PLC and MiniLSS.

To change the value, move the selection bar to the desired line with the  and  keys, and press . The LR area size will be changed in

MiniLSS. If MiniLSS is connected to a PLC a prompt will appear, asking for confirmation to change the values in the PLC as well.

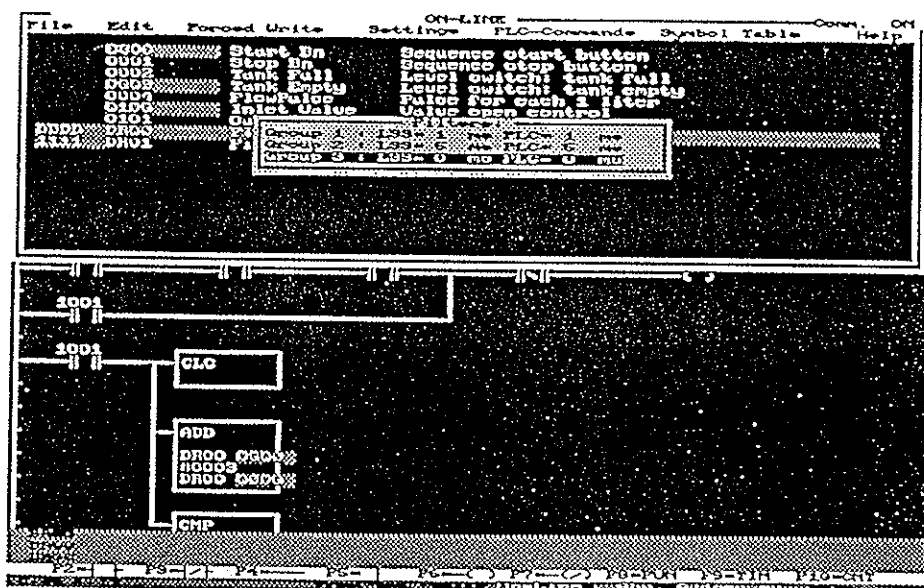
To exit the LR Area definition window press **[Esc]**.

The LR Area definition window can also be accessed from the Diagram Editor by selecting *LR Size* from the *PLC* menu.

6.3.5. Setting Filter Values

The input filter values can be set from the Multimonitoring Page as well as from the Diagram Editor. In the Diagram Editor select *Filters* from the *PLC* menu. In the Multimonitoring Page select *Filter Values* from the *Settings* menu.

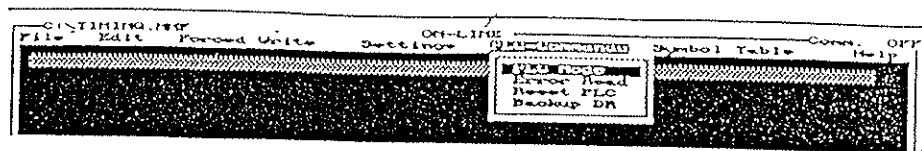
Both selections will produce a filter definition window on the screen:



To change the filter values, move the selection bar into desired group with the **[↑]** and **[↓]** keys. The value can be changed with the **[→]** and **[←]** keys, which scroll all valid values. When the values of all the groups are set, press **[Enter]** to actually change the values. If the PLC is connected to MiniLSS a confirmation is asked for before updating the filter values in the PLC.

6.3.6. Other PLC Commands

the *PLC Commands* menu on the Multimonitoring Page contains commands for setting the PLC mode, reading errors from the PLC, resetting the PLC and backing up DR memory:



The PLC mode can be set to either *Program mode* or *Run mode*.

Error read displays errors currently found in the PLC.

Reset the PLC command clears the the PLC memory.

Backup DR command transfers the data in DR area to EEPROM in the PLC.

6.4. Communication Errors

Monitoring requires continuous communication between the PC and the PLC - the PC queries about states of the PLC's addresses and the PLC responds. Occasionally data received from the PLC will have been altered during transmission. The user will be informed of these communication errors when detected, but no action is then required from him. The message will be retransmitted and probably received the next time without errors.

7.1. Compiling Diagrams

In the PLC the program is in executable binary format. Before a program of diagrams and instruction lists can be downloaded to a PLC, it must be compiled to that format. Using a programming device, the program appears in *Mnemonic List* format, or as a list of instructions terminating in an END instruction. In mnemonic list format there is no division into instruction list elements or diagrams. Diagram comments and address symbols are not contained in the mnemonic list level. MiniLSS uses the mnemonic list format as an intermediate level between diagrams and executable binary format. Although you probably never need to work with mnemonic list format, this file is available as <name>.MNE file on your disk.

The syntax of diagrams is checked during compilation. As the compilation is a two-phase procedure, syntax checking takes place in both phases. Syntax errors will terminate the compiling.

The following syntax checking is made:

- There are not any undefined symbols in the program.
- Every contact and output has a valid address.
- Ladder diagrams are logically consistent.
- Functions, counters and timers all have operands.

The following syntax checks do not terminate the program but prevent downloading:

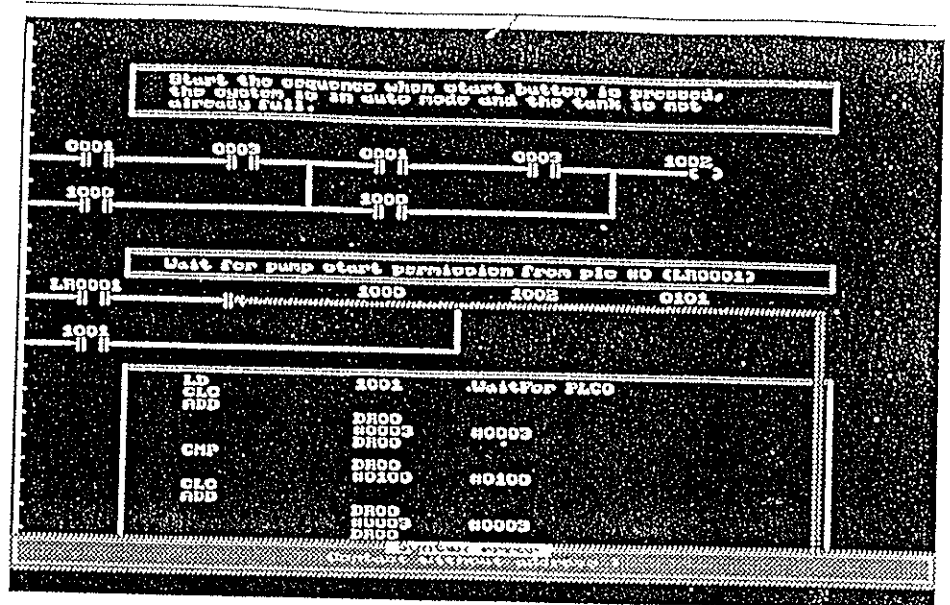
- There is no END instruction at the end of the program.
- The program does not exceed the maximum size of the selected PLC type.

The following syntax checks do not prevent downloading:

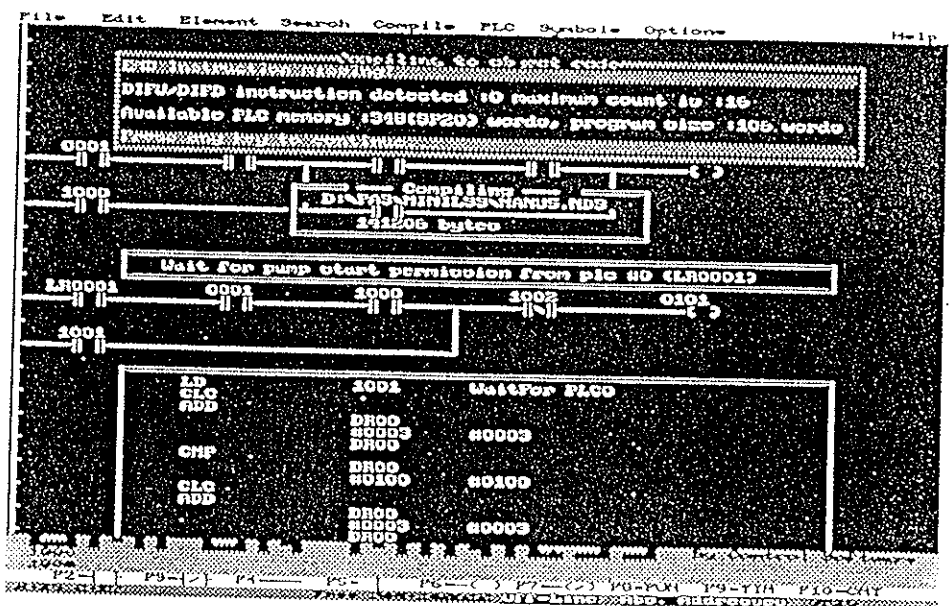
- The number of DIFU/DIFD instructions do not exceed the maximum limit.
- IL/ILC check: for each ILC instruction there is at least one IL instruction.

To begin compiling select *Diagrams->OBJ* from the *Compile* menu (or *Compile ->OBJ* in FILE MNGT menu if in LSS mode). Open the program before compiling, as MiniLSS always compiles diagrams found in the working space.

If a syntax error is found during the first-phase compilation, the compilation terminates and the syntax error is displayed:



The syntax errors found in the second-phase compilation will be displayed in a window:



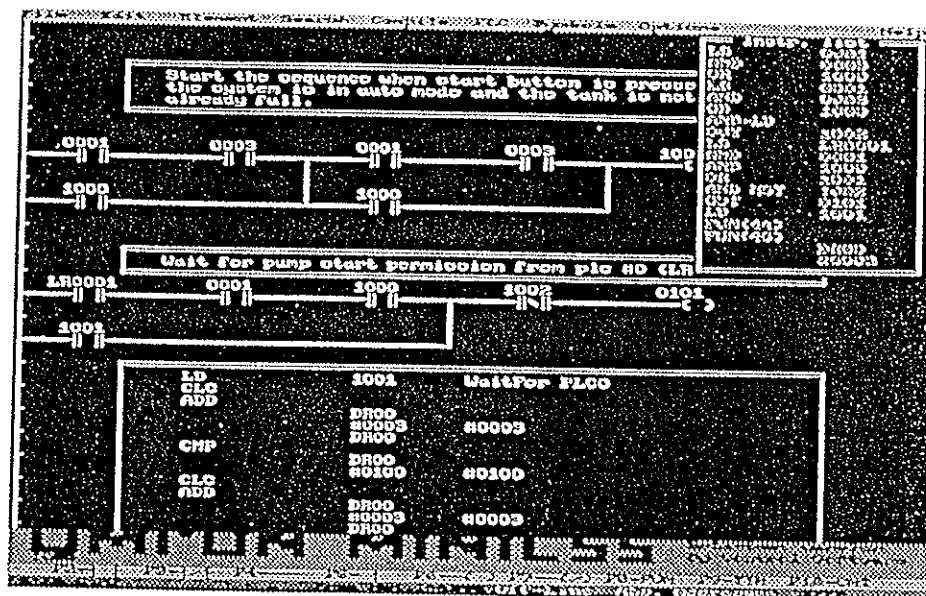
If no syntax errors are found, an executable object file is created. The name of the file is the program's name and its extension is .LOB. This file can be downloaded to the PLC.

There can be diagrams that cannot be compiled to mnemonic format using only addresses assigned to contacts, but a temporary saving of intermediate results is required. MiniLSS uses bits 0108 - 0115 for this purpose. In the default symbol table they are named TempRelayLSS. Do not use these bits directly.

If, for some reason you have to use any of addresses mentioned, change their symbol name. This prevents MiniLSS to use them. You should always leave at least three temporary bits for MiniLSS. If the complexity of diagrams

require more temporary bits, you will get a compile error. In this case assign more temporary bits by setting the symbol text equal to TempRelayLSS. Note that the symbol text is case sensitive. All temporary bits have to be in word 01.

You can compile diagrams to a mnemonic list and view the results on the screen. Select *Mnemo To Screen* from the *Compile* menu. The first-phase compilation is executed and the result is displayed on the screen;



You can use this command to check the syntax of your diagrams without having to compile to object code.

7.2. Compiling Object Code to Diagrams

An object file can also be uploaded from the PLC and compiled to diagrams. When compiling, all logically consistent portions are compiled to ladder diagrams and the rest are put into instruction lists. Since the object file contains no diagram comments, they won't appear in the diagrams. If the program is originally created with MiniLSS, avoid overwriting it by the compilation of object code to diagrams.

7.3. Comparing Programs

If you are uncertain about whether a program you created with MiniLSS is exactly the same as the corresponding program in the PLC, e.g., if you have made small changes with the programming console without updating your program files, you can verify the differences between the programs in the PLC and programs in MiniLSS. First upload the program from the PLC; this produces an object code file with the extension .LOB. Load the diagrams from disk and select *Compare* from the *Compile* menu (or COMPARE from

the FILE MNGT menu if in LSS mode). MiniLSS then prompts you to select the object code file against which the diagrams should be compared. After the file is selected, MiniLSS compiles the diagrams and the object code to mnemonic list and compares each mnemonic code line. If mismatches are detected, the comparison stops and an error message is given with the position where the mismatch was found. If the programs are identical, a message is displayed.

If mismatch is detected, you can view the compiled mnemonic list from the diagrams by selecting *Mnemo To Screen* from the *Compile* menu, and the uploaded mnemonic list by holding the **[Alt]** key down and pressing **[U]**. Modify your diagrams and recompare.

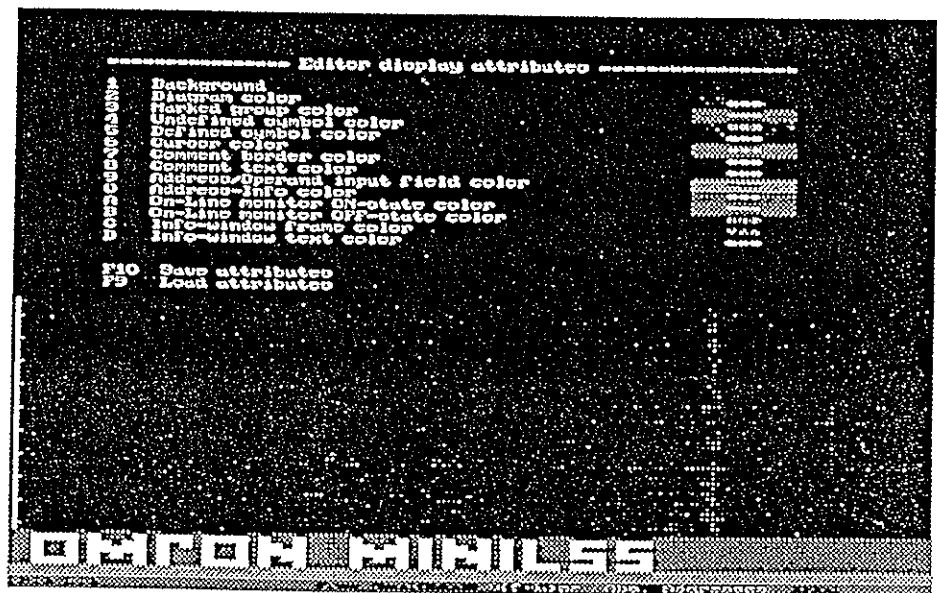
Note that uploading and comparing programs is not needed if all changes to your programs have been made with MiniLSS.

SECTION 8

Customizing Display Colors

Display colors (or attributes) can be customized. The need to change these attributes may arise when using monochrome LCD or plasma displays which are common in Laptop and Notebook PCs.

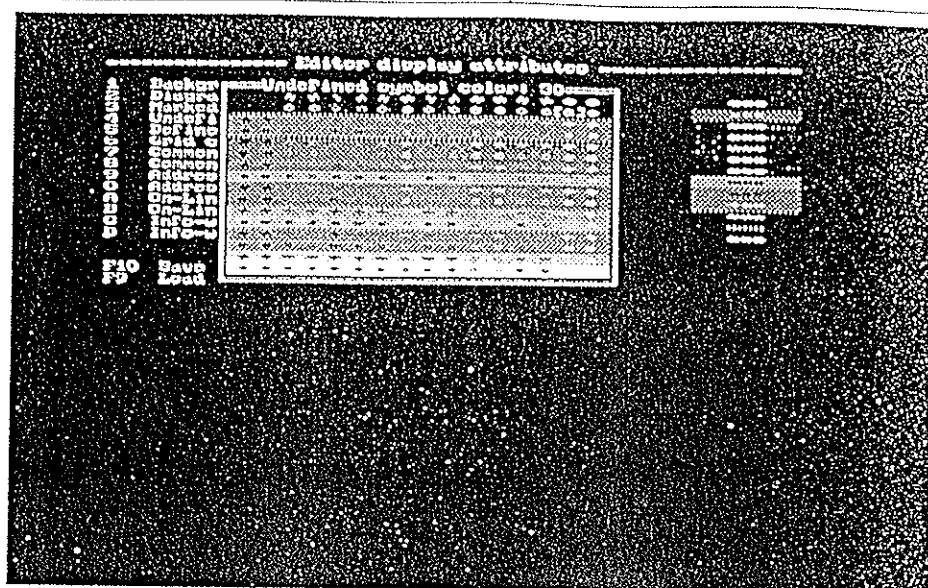
In each editor, pressing **Ctrl****Q** and **C** leads to a display attribute definition screen which appears as follows:




The color selection displays can also be reached from the menus. In MiniLSS mode select *Colors* from the *Options* menu, in LSS mode select *Screen Colors* from the *Setup* Menu.

The items and number of items depends on which editor you were in before going into the display attribute definitions.

Select the item you want to alter by pressing the number/letter which is displayed to the left of the item. A window opens where all possible attributes are shown:



The current color selection is displayed with brackets []. You can change the attribute selection by moving the brackets with cursor keys and accepting the selection with .

You can save attributes to disk and they will automatically be loaded when MiniLSS starts, or you can load the attributes from disk which will override any attribute changes that you have made since the last attribute save operation.

INDEX

- Address, 29
- Address Comments, 15; 16; 55
- Address Exchange Editor, 29; 40; 45
- Addressing modes, 6; 21; 44
- Clearing, 25; 39; 44
- Clipboard, 21; 28; 29; 38; 49
- Compiling, 9; 21; 41
- Compiling Diagrams, 59
- Copying, 24; 39; 44
 - address comments, 16; 46; 47
 - converting diagram format, 25; 26; 27
 - diagrams, 21; 22; 23
 - symbols, 16; 46; 47
- Cutting, 22; 24; 25; 39; 44
- Diagram Comments, 11; 17; 40; 44; 59
- Disconnect, 51
- Display colors, 63
- Downloading, 59; 60
- DR backup, 58
- Edit line, 6; 9
- Editing instruction list, 11; 14
- Editor mode, 6; 9
- Editor screen, 5; 6
- Elements, 40
 - deleting, 8; 10
 - drawing, 7; 9
 - instruction list element, 11; 12; 13; 21
 - moving, 9; 11
- Files, 15; 20; 28; 29; 35; 38; 43; 48; 50; 55; 59; 60
- Header, 35; 36; 40; 49
 - special codes, 35; 36
- Input Filters, 42; 46; 57
- Justified Outputs, 9; 11; 44; 50
- Ladder elements, 7
- LR Area Size, 20; 42; 46; 56
- Marking, 21; 22; 23; 24; 39; 44
- Mnemonic List, 59; 61
- Multimonitoring Page, 55
- Multimonitoring Pages, 51; 53; 54
- Online help, 6
- Orientation, 32
 - landscape, 32; 33; 34; 35
 - mixed, 32; 34; 35
 - portrait, 32; 33; 34; 35
- Page length, 35
- Parallel port, 32; 42; 50
- PLC errors, 58
- PLC mode, 58
- PLC Number, 51
- PLC type, 6; 20; 41; 50
- Printers, 32; 42; 50
- Printing, 39; 49
- Searching, 41; 45; 46
- Serial, 42
- Serial port, 50
- Shortcut keys, 7; 9
- SP16STD.STB, 15
- Status Monitoring, 51
- Symbol, 46; 47; 52
- Symbol Table, 15; 16; 42; 44; 49; 51; 53
- Symbols, 13; 14; 15; 16; 21; 29; 42; 55; 59; 61
 - undefined, 16; 20; 21; 59
- Syntax rules, 8; 10; 59
- TempRelayLSS, 61
- Uploading, 61
- Using monochrome display, 5; 63