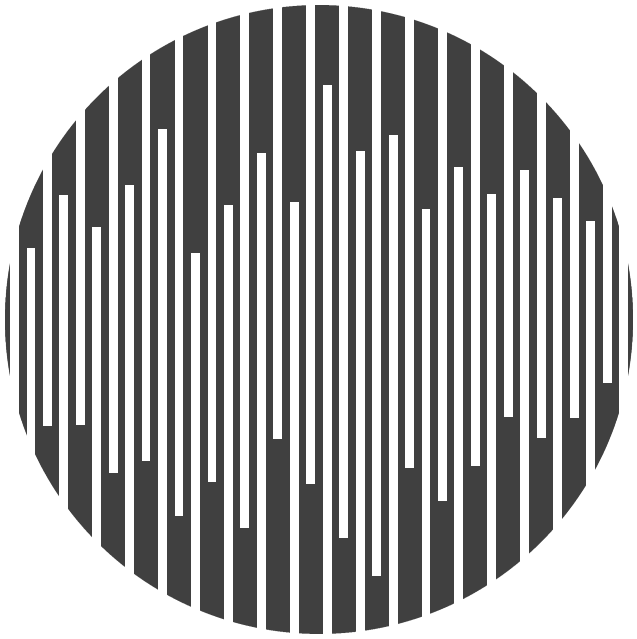
Modsoft Programmer User Manual

890 USE 115 00



GROUPE SCHNEIDER

■ Modicon ■ Square D ■ Telemecanique

MODICON

Modsoft Programmer User Manual

890 USE 115 00 Version 2.6

July

Preface

This is your notification that the software products being supplied to you are subject to an executed software license agreement.

Under the terms of that agreement, the proprietary software you have received cannot be copied, reproduced, transferred, or licensed to anyone but you (unless specifically authorized by the Software License Agreement). If the equipment becomes inoperative, you may transfer the software to a back–up system.

The software media is warranted to be free from defects or workmanship for one year. If warranty work is required, obtain a Return Authorization or Credit authorization from Schneider Automation, Inc. and return the software media postpaid. Schneider Automation, Inc will repair or replace the media as appropriate. Schneider Automation, Inc has available a software maintenance service to provide you with ongoing software support.

This notification is a summary of the obligations relating to the proprietary software being provided. For details consult the executed Software License Agreement.

FAILURE TO ADEQUATELY PROTECT THE PROPRIETARY SOFTWARE IN YOUR POSSESSION COULD RESULT IN BOTH CIVIL AND CRIMINAL LEGAL ACTIONS BEING INITIATED.

The following are trademarks of Schneider Automation, Inc.:

Modicon) P190 984A 984–380 984–381
E984–381 984–385 E984–385 Micro 984 984
984B 984–480 984–485 E984–485 984–680
Modbus) 984x 984–685 VME E984–685 984–780
E984–785 L984–785 984–785 984–A120 Quantum 984–A
130 984–A145 984–351 984–455
Modsoft) Modbus Plus Modsoft) Lite BGAT
M1 Momentum
IBM) is a registered trademark of International Business Machines, Inc.; IBM PC is a trademark of International Business Machines, Inc. DOS{ is a copyright of both

Contents

Chapter3
Main Menu Functions 69

Chapter 1
Introduction

- Purpose of this document
- Programming components and standards
- Loading Modsoft ®

1.1 Purpose

Modsoft 984 Programming Software for DOS serves as an integrated tool for programming, testing, and documentation of Modicon Programmable Logic Controllers. The applications include graphic and text editors, configuration utilities, facilities for transferring code and data to and from the controller, documentation, and online help for the user. The general philosophy is to have a set of source code editors for programs and symbolic information. These source programs consist of ladder logic (LL) operations. Documentation provides programs that are accompanied by comments and reference symbols. By using a library facility, created functions can be reused. Therefore, you achieve an extendible software tool that encourages a top-down programming style for controller applications.

This document is only part of a set designed to support various user needs and levels of expertise. The Modsoft documentation consists of:

- Modicon Ladder Logic Block Library- Use this document to familiarize yourself with controller programming specifics and installation data. 840 USE 101 00
- Quick Reference Guide - This card allows you to quickly see what commands are provided at the key menu entries, including use of quick key entries to save keystrokes. Furthermore, it provides quick introductions to the major aspects of Modsoft. (Included with this manual)
- Modsoft Quick Key Template - A User assist that mounts to the keyboard and lists the most commonly used functions.
- This User Manual - This document is presented in the order of functional use when used for developing, loading, and documenting a Modsoft application program in the field.

1.2 Overview

The Modicon Programmable Controller (PLC) is a computing machine that has its own Central Processor Unit (CPU) i.e. conventional Micro-processor and is equipped with application file Random Access Memory and operating system Read Only Memory (Executive).

Applications development is provided for by another computing hardware unit called a programming panel with the same fundamentals as above except the operating system instead of being a custom executive is DOS. The P230 Programming panel can put data in or take data out of the controller given the execution of the proper application software.

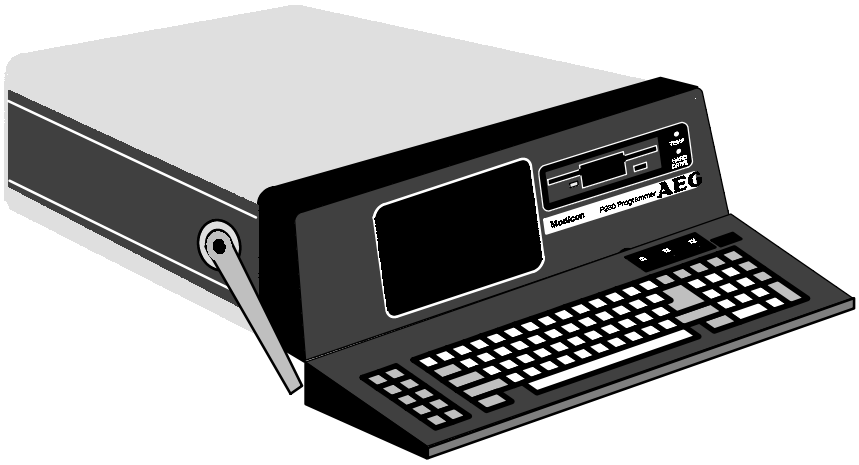


Figure 1 Modicon P230 Programming Panel Front View

As you program, you will want to keep in mind your current pathway. As stated above you have two independent computers and “one” screen. The modes of operation available to you allow you to look at the screen as a window into the controller’s logic or the panel logic and includes the ability to change data. There are aids to help to identify the current operating mode and various levels of automated file handling and user messages to eliminate errors are provided.

1.2.1 File Related Devices

You can load files directly from panel accessory devices (disks) to the Controller. The files are first brought into the panel memory then transferred to the controller. When working with the devices you identify them as A: for the floppy drive (external access) and C: for the hard drive

(internal access). Given the use, there are various paths and data handling you must consider. In general the panel offers:

- Panel communicating directly to the Controller RAM. Modification to files is done within the Controller RAM.
- Panel not communicating with the controller but file modifications are being made in the panel. The Controller can be executing its own program on a stand-alone basis. All modifications to programs are considered temporary and must be saved to the floppy drive or hard drive or they will be lost by file overwrite or Panel power down.

1.3 Operating Modes

When using Modsoft application software, your functional requirements and file usage are supported by the following modes:

- Offline
  Online
  Combined

Modsoft displays a mode indicator in the upper right hand corner of the screen, between F8 and F9, to help you recognize the current operating mode. The prompts used are:

OFF for Offline
ON for Online
Combo for Combined

An additional assist is provided after the F9 where a number, in the range of 1 to 247, is displayed. This number is the PLC slave address that you will set in the Offline "Change PLC Address" or Online "Direct to PLC" communication parameter screen and is displayed to indicate a login condition. The letter preceding the number is either:

R for PLC ***R*** unning or
S for PLC ***S*** topped.

When Modsoft is run you can set a User Level indicator (0-8) between the F7 and F8 by using the /L switch or setting the `user_level=` in the Modsoft.UPF (User Preference File). Details on these two options are provided in the "User Defined" Appendix of this document.

Here is where these fields are displayed:

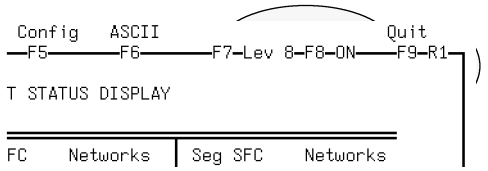


Figure 2 Operator Assist for Mode and PLC Address

If you have chosen the "State Flow" function, the word STATE is displayed between F2 and F3. In addition, the current panel program `name` (ENV) is printed at the F2 location on most Offline screens.

1.3.1 Offline

This mode allows configuration and program modifications without using the controller as a host. All operations interact between local files and the application. Modsoft controls and provides access to the functions allowed for the mode you are in. Offline is the only mode from which you can set the panel communication ports. Thus while Offline, you make a connection to one controller so that an Online session is guaranteed to work on that controller only. Available in Offline only are:

Select Program
New Program
Save Changes
Save All Files
Save as...
Change PLC Address
Program File Settings
Location Of Program Names

1.3.2 Online

In this mode (previously known as Quick Mode) the application communicates with the controller in a condition called “attached” or “logged in”. All changes created at the panel are reflected *in the controller* ONLY. In addition, comments and/or symbols changed can be used by choosing the Select Program option. Your Menu options are:

Select Program
Direct to PLC

1.3.2.1 Combined

The Combined mode provides the means for debugging a program during the installation phase, where any logic changes made in the PLC should be saved in the offline file. During this operation, changes that are made to modify the controller are also made in the temporary files residing in the panel. The files can then be saved without the need to upload the program from the controller.

1.4 Programming Components

From the main menu you can step down a menu hierarchy and select options for each sub-menu. In parallel, you can access the underlying data to which your program refers. The main menu looks like the figure below. In this illustration you can see a pulldown menu that provides options under "`Tools`" and you can see the Tracking help line (lowest line) that provides a brief definition to assist you regarding the functionality .

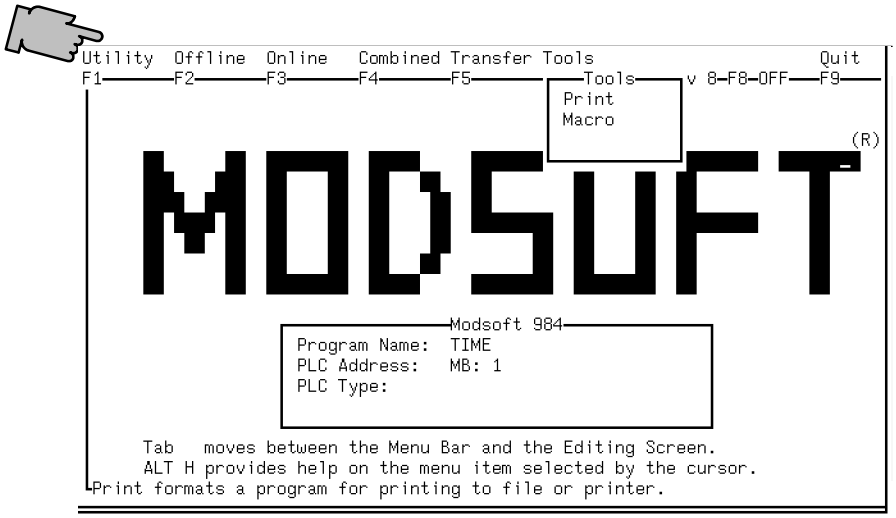


Figure 3 Main Menu Selection Line

`Utility` `Transfer` and `Tools` combine common functions and are available in the menu header of most other selected functions. The main menu also displays the Three programming options `Offline, Online` and `Combined`, which have associated configuration and supporting editors. The Program Name assignment automates loading and saving of complete programs, symbol tables, libraries, etc. including communications parameters with ***ONE*** command.

A program consists of the Source Level Program with underlying networks (program source), the Reference Symbol File that assigns symbols to reference numbers used in the program (.RFS file) and the Program Comments File .PCM. The network symbols and comments are kept in separate files for the same program. This separation helps in a translation effort where the text can be provided in different languages.

1.4.1 Hardware/Software Requirements

The minimum hardware required for Modsoft is:

Hardware: 486 CPU
Hard Disk.

Modsoft distribution disks are Double Density 3 1/2 inch media.

Note Due to the memory management required to maintain more than 30 files open simultaneously, Modsoft 2.4 and up needs to be run with MSDOS Version 5.0 or greater, and under Windows 3.0 or higher if using windows.

However, MSDOS 5.0 with EMM386 and the NOEMS switch can cause problems not experienced if using MSDOS 6.0 (See Release notes for more detail).

Tip Modbus RTU mode may demonstrate problems on Windows 95, Windows NT or OS2 and may experience difficulty in systems with active network adapters or multi-tasking.

1.4.2 Conventions

Throughout this document, when you are directed to enter a keyboard command, it is set in bold type, e.g., **A:INST ALL** Unless otherwise directed, commands are invoked by pressing the ENTER key (same as RETURN key on some keyboards and symbolized as ↵).

Specific keyboard keys appear as on the keyboard, in brackets and bold type, e.g., <**Esc** >

Keys represented by graphic symbols are spelled out, e.g., <**backspace**>, not ←

Key sequences with *Alt* or *Control* specified mean you hold those keys down while also pressing the second referenced key i.e., <**Alt H**> means hold the Alt key down and press H.

Direct references to display menu prompts are enclosed in quotes, or printed in Times Font as in “Info”.

Some menu selection lists are identical for different functions. Any entry NOT available is colored Red or shaded Grey depending on your monitor.

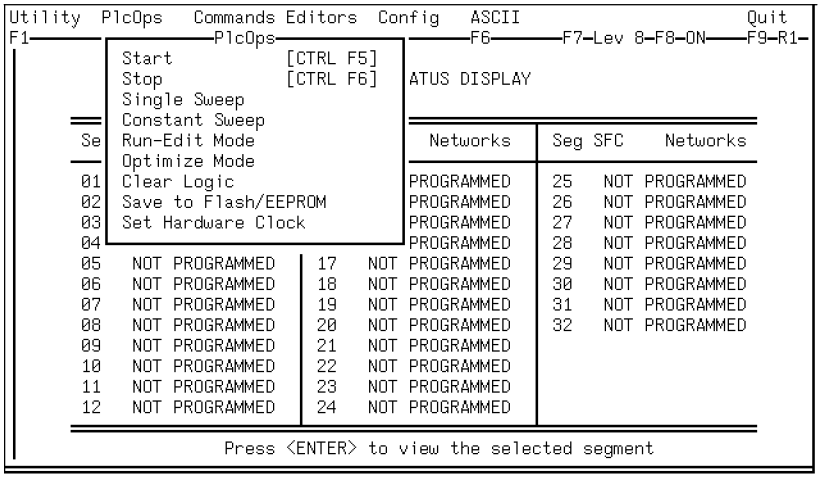


Figure 4 Example of Online PLCOPS Not selectable entries

1.4.2.1 Miscellaneous

You can construct your company logo on Modsoft main menu screen by editing the modsoft.sys file located in the Modsoft runtime directory.

If using a color monitor, four color presentations are provided as prompts. They are:

The menu bar - Black on White
Editors are - White on Blue
Message Window - Black on White
Error Window - White on Red

1.4.2.2 Modsoft CommandLine Switches

Command Line switches are entered using the **/** or **-** prefixes. Some switches also take a parameter immediately following the switch. Switches may be placed anywhere on the command line after the **Modsoft** command and may be in any order. Switches are not case sensitive. If there is a conflict the last switch takes precedence. The .ENV filename may also be given on the command line, and if given that file is automatically loaded.

To use these flags, type **modsoft** followed by a space, slash, and the following letter ***in upper or lower case***.

Example: MODSOFT /o

CommandLine Switch Summary:

/d Allow download of duplicate coils

/i IEC Standard characters are not forced. This switch allows other characters to be used for symbols.

/j Modsoft.jou Keystroke journal file is created only if this flag is used.

/ln Takes the number “n” to set the default User Access Level. This is a number between 0 and 8. Each incremental increase includes all privileges of the levels below it.

- 0 Allows monitoring only.
- 1 RDE Online Changes.
- 2 Allows PLC to File transfer.
- 3 Allows PLC Start, Stop.
- 4 Allows File to PLC transfer.
- 5 Save Programs, Offline editors.
- 6 Allows Full Programming.
- 7 Same as level 6.
- 8 Allows Executive Downloads.

/rn Read keystrokes from the file MODSOFT.JOU, (if elected)
(Where n = seconds to delay between reads and is primarily used for demonstrations.)
The n is optional and has a 0 default value.

/o This switch takes a command parameter and performs an immediate execute of the command. If no command is given the default is to go immediately to Online Mode. These command functions override User Preference level settings and when completed Modsoft Exits. (You must have previously set up a .ENV file by starting Modsoft and exiting normally).

Commands are:

Offline	Go to Offline Segment screen
Online	Go to Online Segment screen
Combined	Go to Combined Segment screen
Status	Display Status Screen
Verify	Verify File and PLC
File_to_PLC	Download file to PLC
PLC_To_File	Upload file from PLC
Print	The Print document function

/s Do not delete temporary files before re-entering Modsoft.
This option is necessary if you did not save to a file (in the previous session) before leaving Modsoft. For example, if a power failure occurred,

then after restarting the panel with Modsoft /s you will continue from your previous session (you should do a SaveAll from the “Offline”).

/vnn Allows changing the Modbus Plus software interrupt vector number where: nn = 01 - FF (Hex). (See Modicon IBM Host Based Devices User’s Guide 890 USE 102 00 for further information about setting the software interrupt).

/u Disable the creation of the “Used Reference table” in Combined mode. Using this switch saves time during the Convert process.

/5 5 digit reference 16 bit processor (is the default for all controllers without Extra Registers)

/6 6 digit reference 24 bit processor (is the default for all Extra Register configured controllers)

☞ **Note** To properly support the builtin 16 bit math functions (AD16, SU16, MU16 and DV16) ***in the PC-E984-685/785*** you must explicitly set the /6 switch to enable 6 digit entries in the function blocks. With this switch set you can use a 5 digit constant by putting the # sign as the first digit i.e., #65535.

/t Save all Communications messages (to and from the PLC) in the file Modsoft.bus. (Used in conjunction with /e engineering mode).

☞ **Note** For information on Editing a User Preference file for inclusion at Modsoft start up, see Appendix D. The .ENV file is defined in Appendix E.

If using Modsoft specifically to execute the Print function (a full Print) you can quickly achieve the desired function from the command line by typing:

```
C:\>Modsoft /L6 /OPRINT print_file_name
```

In situations where this function is frequently used, inclusion in the UPF file may be useful but not recommended. Given the UPF entry:

```
immediate_command = Print
```

The execution becomes:

```
C:\>Modsoft /L6 print_file_name
```

1.4.3 Example

During the user orientation (provided in subsequent chapters), an example of a ladder logic program with all its components is used for illustration and is provided by Schneider Automation, Inc. on the Modsoft distribution disk set. The example is called TIME and is a simple control program solution to a requirement to implement an Hour, Minute, and Second timer scheme that could, for instance, be output to a segment LED display. The example progresses from configuration to traffic cop, both of which are downloaded to bring the controller out of the “Not Configured” state. The same example is used in the ladder logic programming chapter, and concludes with its program listing. The “Time” example is found in \MODSOFT\PROGRAMS.

The Controller used to develop this document is a Quantum Controller therefore, Reference values are presented in a 6 digit format.

How you initialize this program is explained after the Modsoft installation procedure.

Note ***Not Configured*** is the controller’s state when first purchased or after having all remnants of equipment configuration removed from memory. The processor has POWER and READY, and therefore has passed diagnostics and is awaiting a program input that tells the processor about its environment. `Not Configured` was previously described as “dim awareness”.

1.5 Installing Modsoft

Before starting Modsoft, if you have a color graphic adaptor and a monochrome screen, you must enter the following DOS Set command:

SET SCREEN=MONO

or use Modsoft.UPF with “screen=1” (See Appendix D)

If using Modbus Plus, this driver also needs to be defined.

You install the Modsoft application software from the distribution media via the disk drive appropriate for your system. The installation is automated and prompts you to confirm your install command. The install procedure checks for the availability of usable disk space before installing Modsoft. If there is not enough space, you are warned.

The following is an example of one way to transfer the software from floppy disk to the system internal hard disk:

Tip: Use “update” if you have a previous version of Modsoft installed.

1.5.1 Install Procedure

Step 1 Power up the system that you intend to load.
Step 2 Insert the distribution diskette (Disk 1) in drive A.
Step 3 Type **A:INST ALL** ↵

The Installation software displays relevant information for you. You can always Quit the Installation process by pressing the <**Ctrl** + **X**> keys.

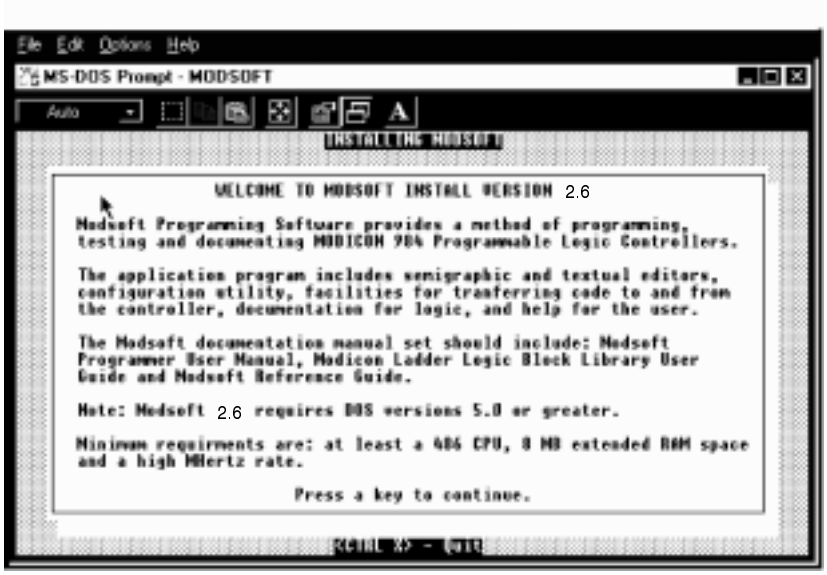


Figure 5 Automated Installation Information Screen

The next display informing you about the disk requirements.

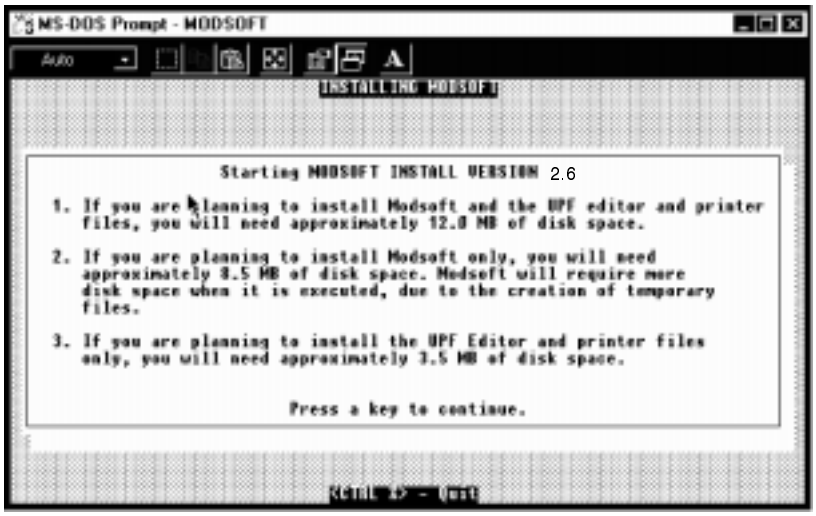


Figure 6 Screen 2 for memory requirements

the following notes are displayed next:

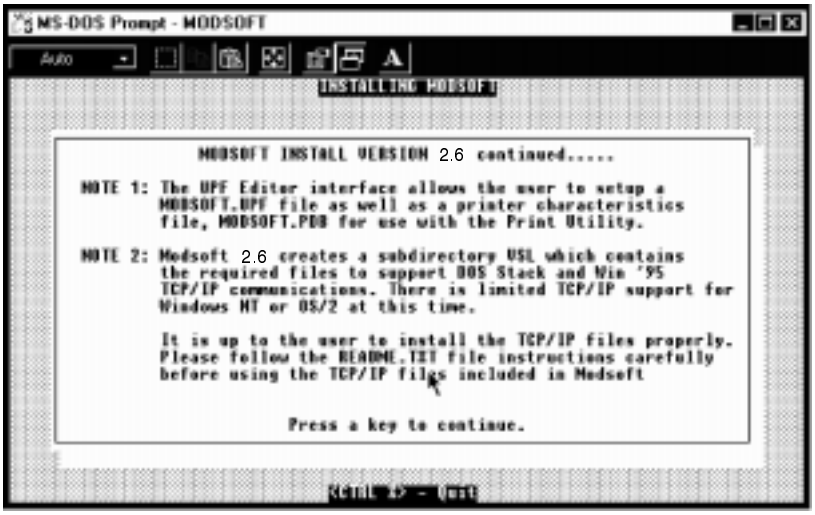


Figure 7 Install Notes

You are provided with a selection list that allows some customizing of the features you want to install. Move the cursor to the item you want and press the return key.

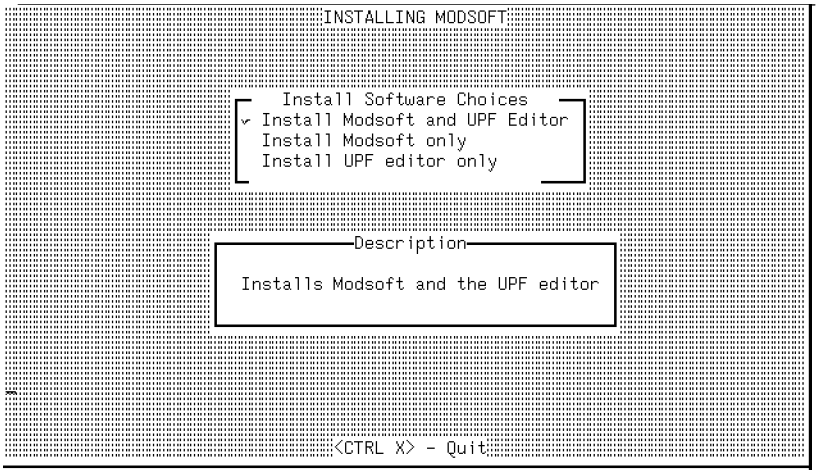


Figure 8 Installation Select Example

Based on the selection you make, a summary screen is displayed to confirm the amount of memory you will need to install the selected option.

As you continue the installation, the drive prompt screen allows you to select the destination drive from the various devices commonly available. A display is provided that gives you the overall memory resources available particular to your choice. Use the cursor <↑> and <↓> keys to move among the available selections and press the return key to make the selection.

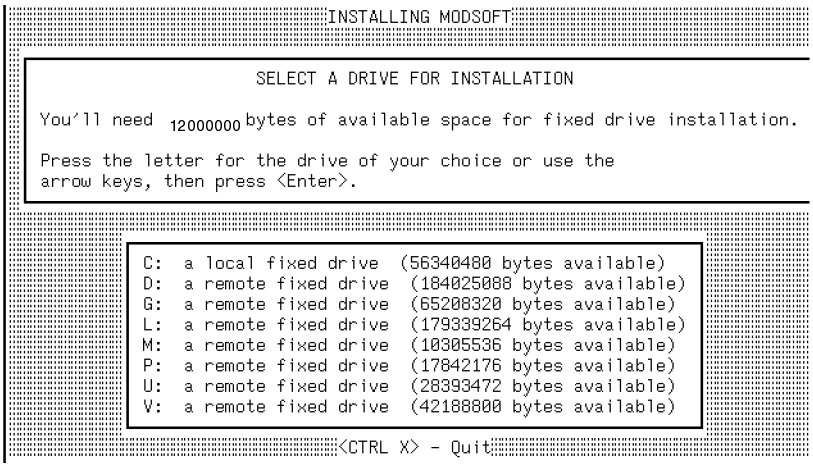


Figure 9 Input Path Select Screen

It is important that the transfer command be correct because the installation process automatically (default elected) creates the directories \MODSOFT, \MODSOFT\TMP, \MODSOFT\RUNTIME, MODSOFT\PROGRAMS, MODSOFT\LOADABLE and MODSOFT\A120IOD on the destination device. The next prompt screen gives you the option to specify the directory name you want. This illustration uses \MODSOFT. It is highly recommended that only experienced Modsoft users change the installation directory from the default.

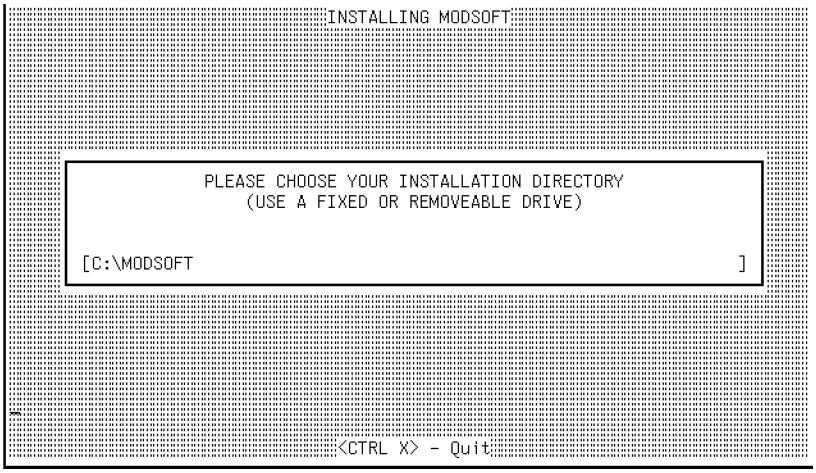


Figure 10 Directory Initialization Screen

If the directory you choose already exists indicating a previous version of Modsoft exists, you will see:

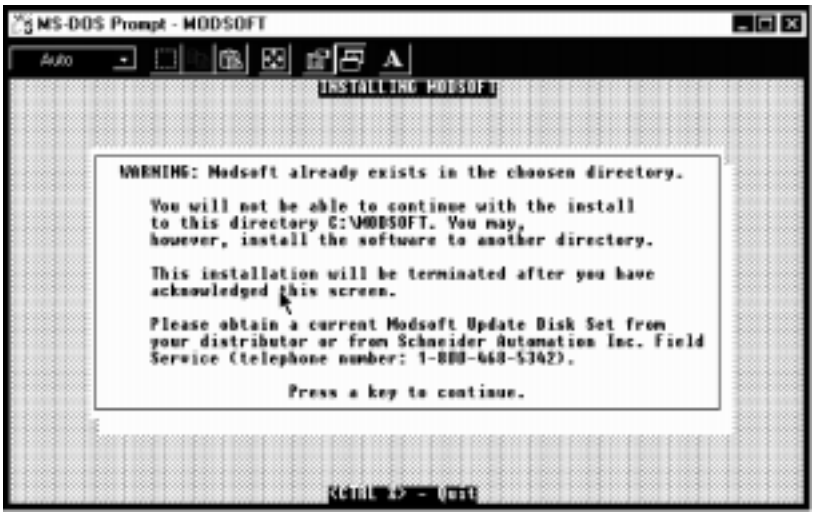


Figure 11 Directory Exists Screen.

After the directory is established, you are directed to proceed with prompts to fill in your name then a second prompt is displayed to enter your company name.

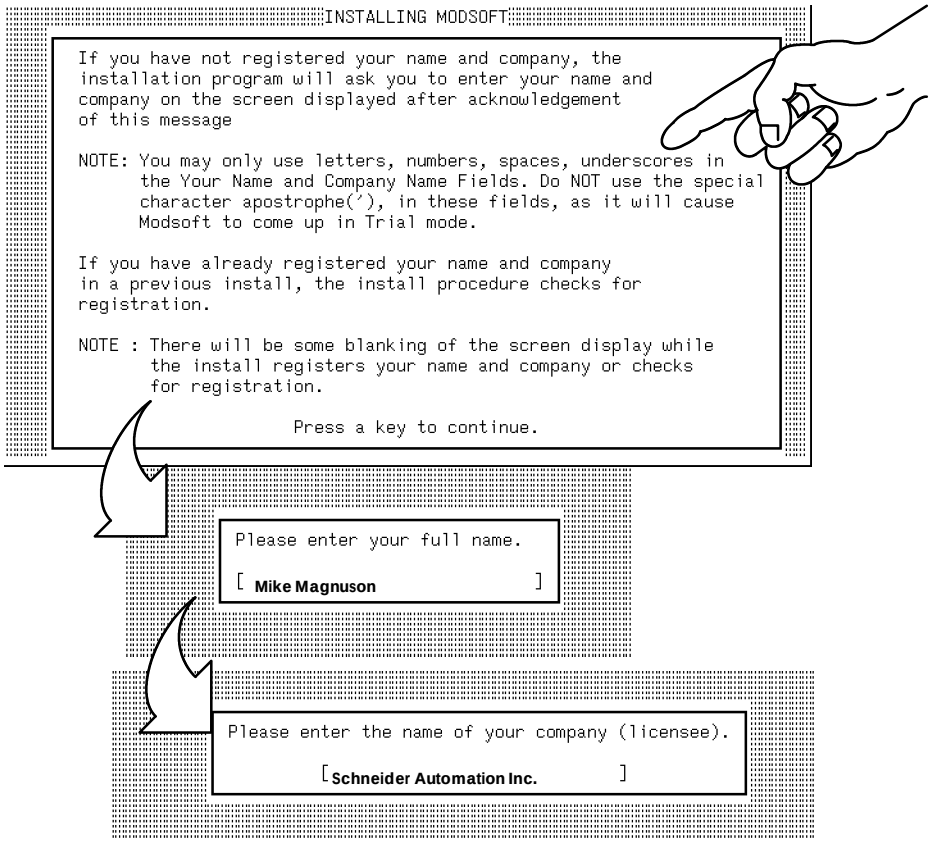


Figure 12 User Name and Company prompt displays

After entering the Company Name The License information display is posted for confirmation:

Tip In some cases where networking software is installed and remains resident (TSR) your install may abort (at this point in the installation). To complete the install remove the offending TSR software, reboot and restart the install.

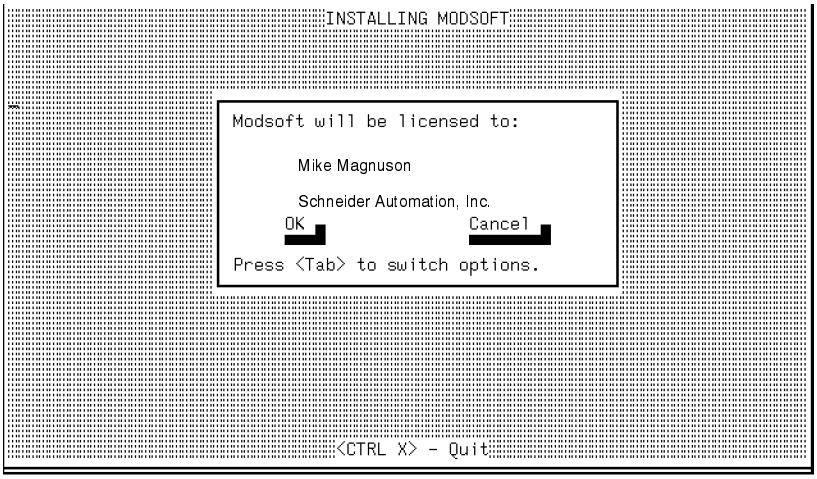


Figure 13 License Information Confirmation Display

As the owner of the registered license you can re-install Modsoft if required; however, others can not modify the registration.

The transfer of files is automatically started when you accept the license data.

The following message is displayed when disk 1 is completed:

Please insert disk 2.
in drive A: strike a key when ready.

This message is repeated to prompt you for each subsequent disk.

During the file transfer process, a display is presented to keep you informed of the process status. File names are listed and a horizontal bar graph is filled in to correspond to the amount of total transfer completed. The graph has two tracks: The top one for files and the bottom one for the transfer queue .

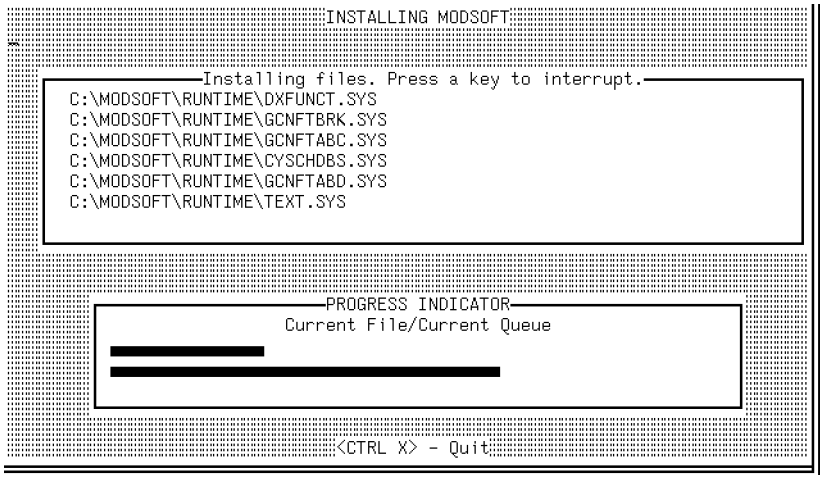


Figure 14 Load Progress Status Display

At the conclusion of a successful transfer (an error is reported if unsuccessful), you are prompted to continue the automated Installation procedure.

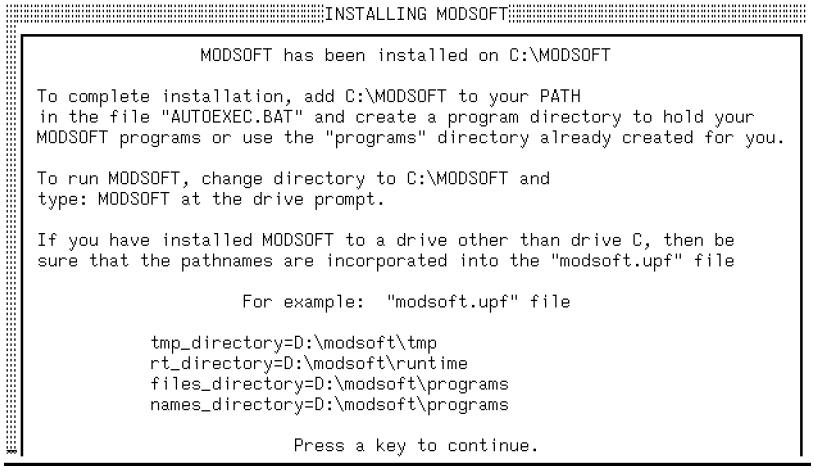


Figure 15 Additional Installation Information

It is recommended that you check the CONFIG.SYS file to verify that the environment parameters FILES and BUFFERS are set=30, as a minimum.

Files=30
Buffers=30

(No spaces in set command)

☞ **Note** If you are running the Modsoft application under Windows you will have to allocate a higher number of files and buffers. Approximately 50% higher should be sufficient ie., 45 files and buffers.

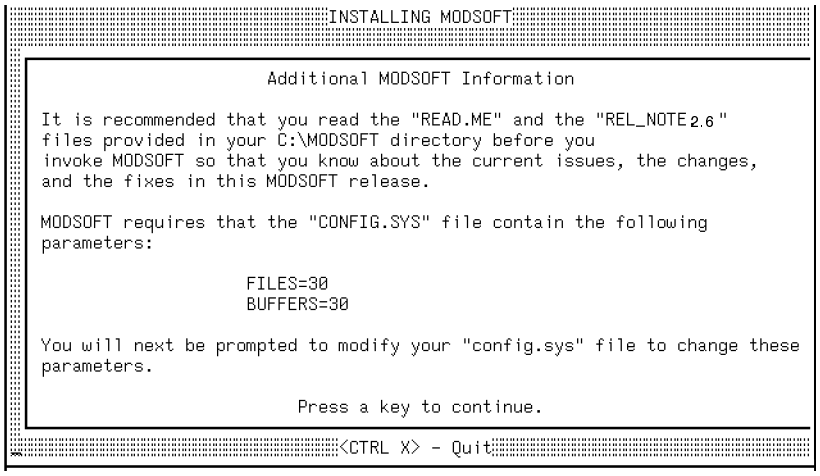


Figure 16 Installation Information display

The procedure will allow you the option of letting the installation software make the change, or allowing you to either make the change or bypass the change if you know the parameters are already set. Your original Config.Sys is saved as a backup file (for example Config.01) if you decide to have the installation change the config.sys file.

Some of the setup requirements can be done automatically if you elect to do so. The illustrations that follow indicate the interaction taken if you elect to have file updates done by the application:

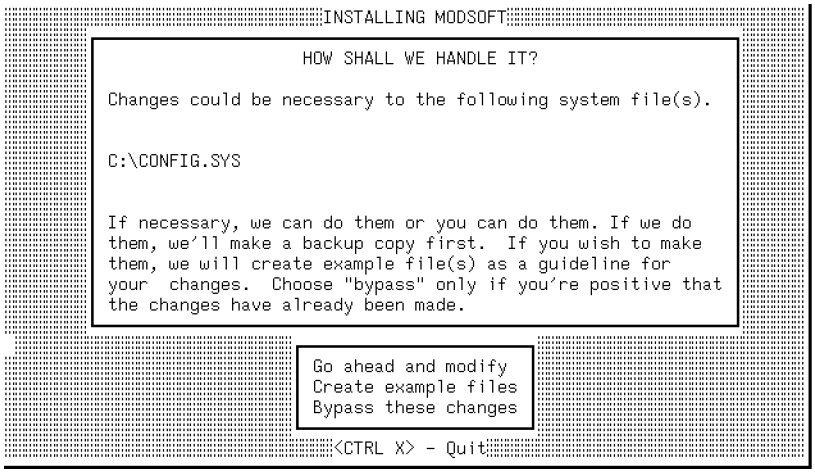


Figure 17 Choosing Assistance

When you ask for modification, another display is provided to request an input for the ***root directory*** and relates to the Config.sys file.

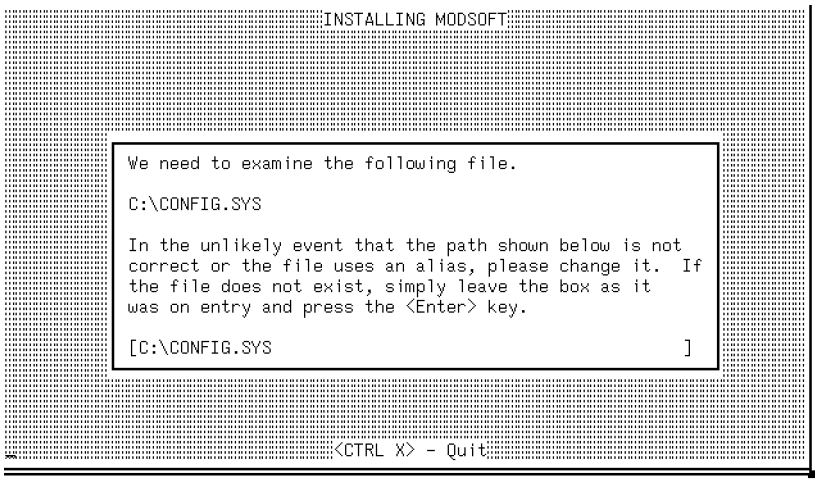


Figure 18 Handling the Config.Sys file

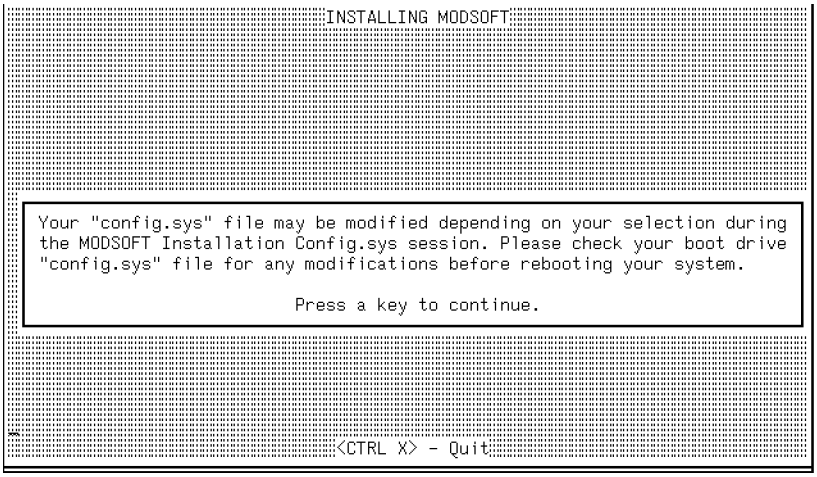


Figure 19 Modification Summary

At the conclusion of the installation process, The done screen is displayed:

Figure 20 Installation Done Display

You should proceed by taking the following steps:

Step1 To implement the start-up parameters you are advised to re-boot the panel. Before re-booting you should ensure that any disk caching software pending writes are complete. Press <**Ctr**l-**Alt**-**Del**> together, or power cycle the hardware.

Step 2 The first step after the Installation, is to start the program by typing:

CD MODSOFT then **MODSOFT** ↵ (***commandswitches are appliedhere***), the result of which is the boot display. This screen provides the user with Product identification, Version number of the software and the user License information.

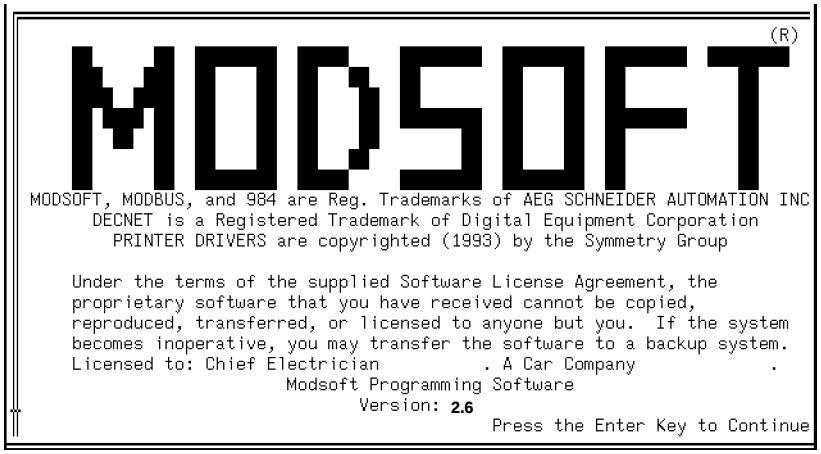


Figure 21 Example Modsoft Application Boot Screen

1.5.2 Install UPF Editor Only

This option is selected when you only want to install the User Preference File editor and Printer data base set up. The disk Memory required is **4** megabytes. The prompt screens follow a similar flow as that presented earlier in this chapter.

1.5.3 Additional Disk

The delivered software contains 1 disk in addition to the Modsoft set. This additional disk contains a number of Loadable functions, Loadable executives and device drivers for A120 I/O. You can put the software routines in Modsoft directories or use them as required directly from your a: drive using the import (Load) facility of the `Configuration → Loadable` Selection.

1.6 Update

If you have a previous version of Modsoft, you should be using an "Update" diskette set. The automated install and user assist displays are ***nearly identical***. Start the process by inserting disk 1 and typing A:Update↵. Two introductory screens are displayed to assist you.

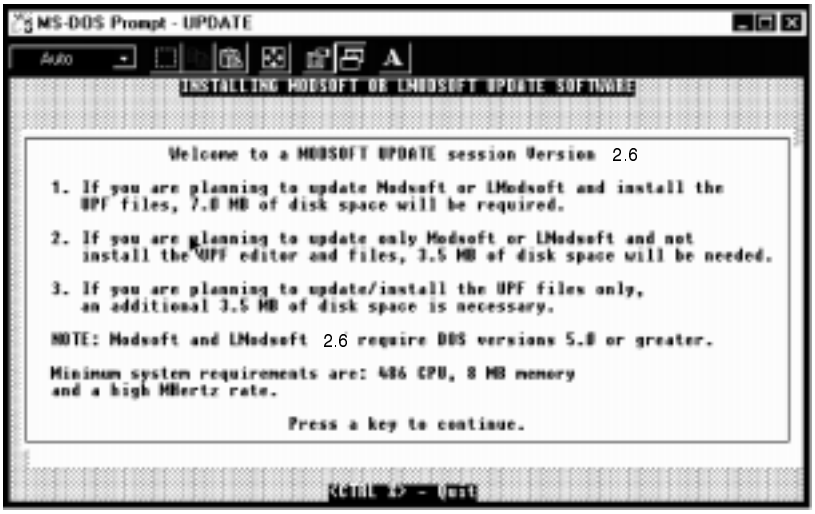


Figure 22 Initial Update session screen

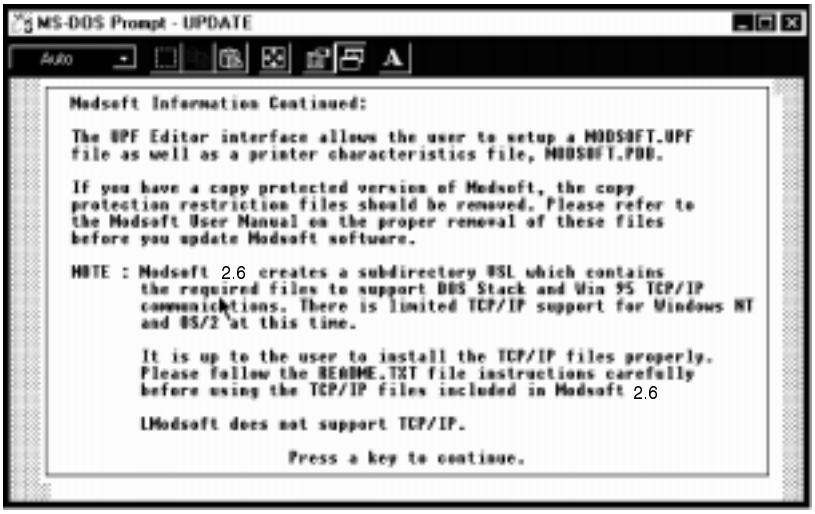


Figure 23 Second page of Update Information

You must select from the Five displayed Update choices. The next User screen appears in the form of the one-time registration information is similar to the ***Install*** Modsoft process previously documented. If you have previously registered, the automated disk loading procedure is started at this point.

☞ **Note** When Updating Modsoft, your Modsoft.pdb (Printer Data Base) is not updated automatically. This is to preserve your current UPF defined printer preferences (see Printer Chapter for More information.

1.7 Programming Quick Start

Now that you have your panel ready to create ladder logic programs, a flow chart of the steps of the process is shown below. The remainder of this document is related to the MODSOFT menu structure and the general process you go through to develop and document your efforts.

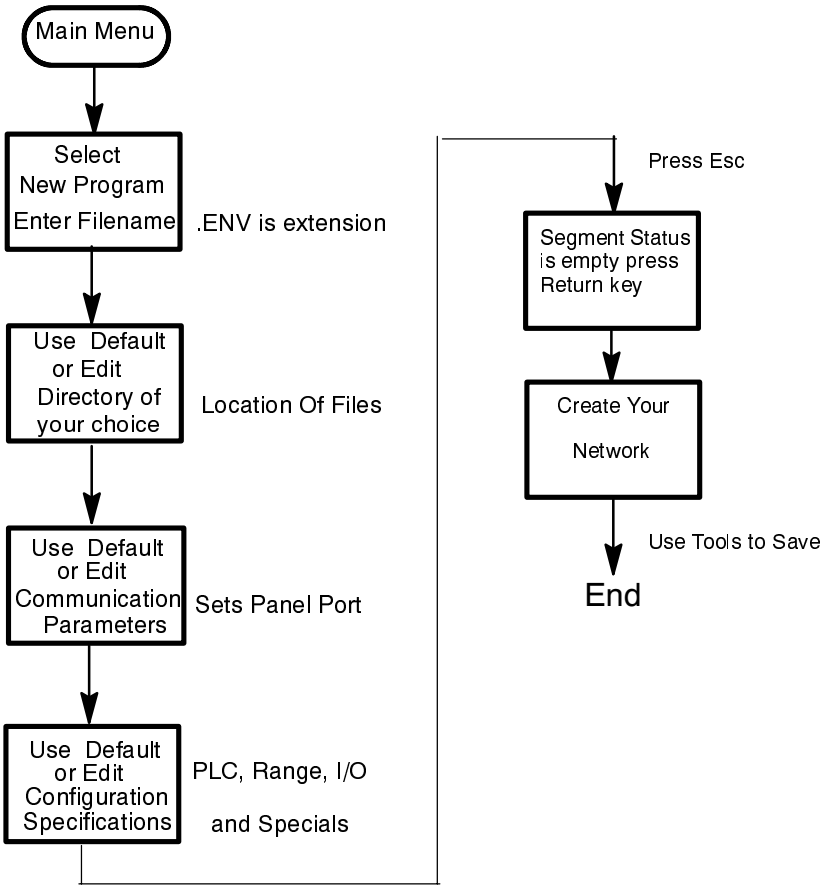


Figure 24 Quick Start Procedure

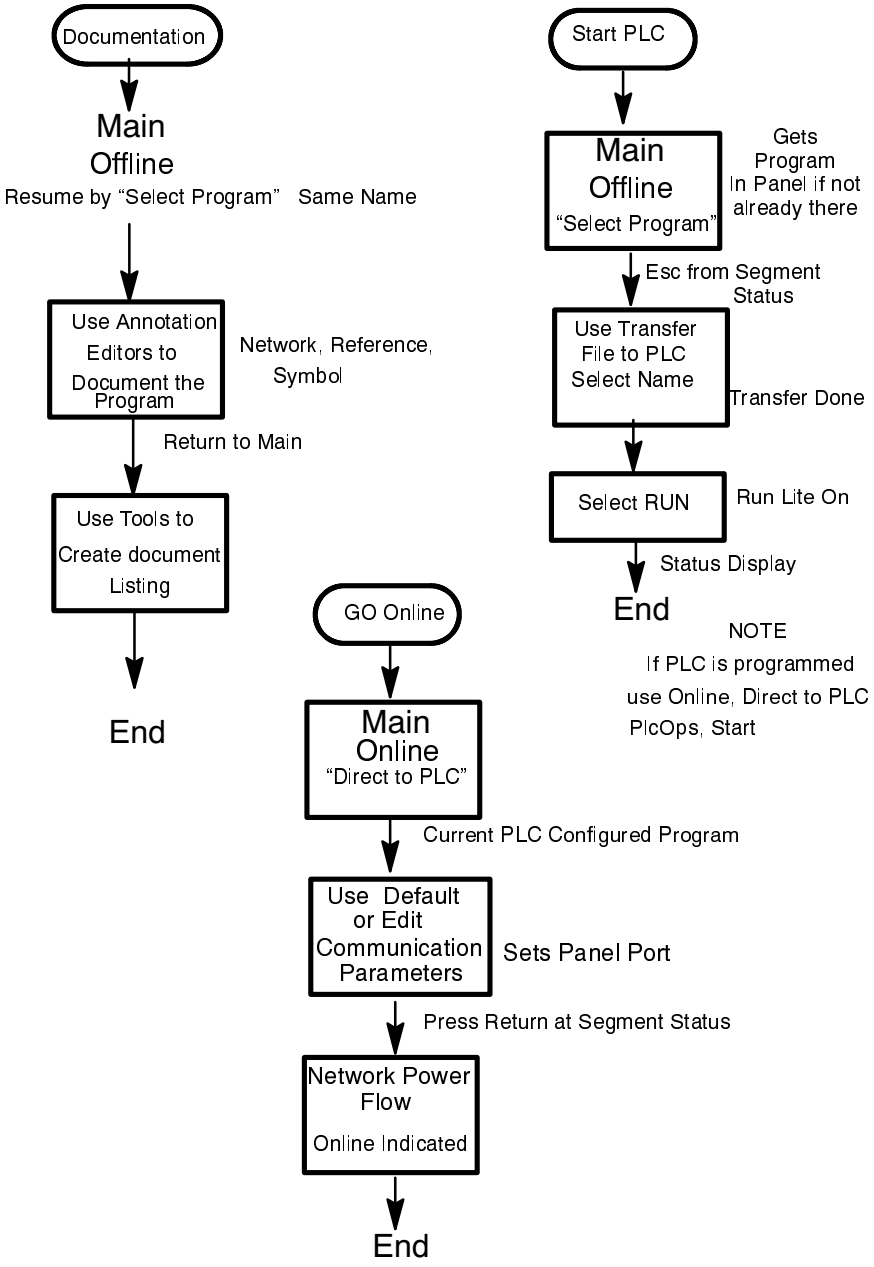


Figure 25 Quick Start Procedure Continued

1.7.1 Initial Start

If you are choosing a startup parameter, you need to use an `MS DOS Prompt` window because you are adding the parameters to the Modsoft executable file command line. If you execute a Modsoft shortcut in windows 95, you can append the startup parameters by putting a ? in the initial file specifier then adding the startup command for that shortcut. For example, if you are going through a keystroke sequence that is to be repeated a number of times, you can set the Journal parameter by typing:

c:\Modsoft /j

Each keystroke you press after the application starts, is recorded in the .Jou file for later play back. You should become familiar with shorthand keys because each keystroke is recorded including menu search cursor movement to find the function you want. Also, while the play back time is variable, 1 or 2 seconds is appropriate to produce animation of the operation. To restart Modsoft using the .Jou file as input file type:

C:\Modsoft /rn (Where n is in seconds)

1.7.2 Starting a spares replacement PLC

It is possible that a PLC with EPROM (or Flash) memory that has been in service can be placed in spare stock in such a way that it will **immediately enter the RUN Mode when power is re-applied.** The potential exists for the program and configuration in the PLC to be executed with some attached I/O being inadvertently activated.

1.7.2.1 Start up precautions

--If the PLC has a ***battery***, it is advisable to ***remove*** it.
--If the PLC has EPROM or Flash or otherwise can retain configuration and program instructions; the PLC should be conditioned to insure it will not automatically enter RUN when power is applied. This can be done by performing a Save to EEPROM (or save to Flash) operation with ***Option selections set so the PLC is stopped after power is re-applied***
--Another way is to ***download an empty program*** (no i/o or program) and save that to EEPROM (or save to Flash) before removing the unit to spares.

1.7.2.2 Startup

--Apply power the the PLC in such a was that the ***I/O is not powered*** or connected. This arrangement allows downloading the program intended for the installation before applying power to the I/O.
--Apply power to the PLC and ***ST OP before connecting any Remote I/O, Distributed I/O or Peer Coppedconnections***
--Apply power to the PLC and ***ST OP before connecting to any Interbus-s, Profibus, Fieldbus etc. I/O*** These may best be disconnected at the I/O head in the local backplane.

1.8 Use of Interim Programs

If you have developed PLC programs subsequent to Modsoft release 2.0, you can use them with release 2.1 (and above) software if you create the now required Environment file relationship. The Environment file must have the same name as the associated Modsoft program file set!

Here are the steps you need to follow:

1) Go to your Program file sets and write down the names of all .CFG files.

2) Also note the path to the program sets.

3) Run Modsoft selecting New Program and use one of the configured program set names you found.

4) Enter the proper path for the file set named.

5) Choose a save if you alter any of the program set parameters.

The environment relation file is now in existence and your programs can be selected from the displayed SelectProgramlisting.

1.8.1 Example

This example demonstrates the above process.

For step 1 you will be at the DOS prompt where you enter the commands illustrated.

```
C:\MODSOFT>cd programs

 9:50:19.63 Wed  4-28-1993
C:\MODSOFT\PROGRAMS>dir *.cfg

 Volume in drive C is DISK1_VOL1
 Directory of  C:\MODSOFT\PROGRAMS

TIME     CFG     5904   4-01-93   2:11p
PIDDEMO  CFG     5700   1-05-93  12:41p
PCFLDEMO CFG     5700  12-09-92  12:46p
AGA3     CFG     5736  12-11-92   9:45a
TIME_LL  CFG     5912   8-04-92   8:24a
        5 File(s)   7008256 bytes free
```

Figure 26 DOS Program Directory path and *.CFG Listing

You must already know where the path was set to list the .CFG files as suggested. If you are not sure, use the Offline "Location of Program Names" to see the drive and path.

To find out if there is a corresponding .ENV use the same DOS command to list them. For example:

```
C:\MODSOFT\PROGRAMS>dir * .env

 Volume in drive C has no label
 Volume Serial Number is 3B55-10CC
 Directory of C:\MODSOFT\PROGRAMS

TIME       .env        1,048 07-21-94  12:23p
TIME785    .env        4,482 09-14-94   1:18p
QASXXXX    .env        3,830 09-14-94  11:41a
584BIN     .env          740 07-05-94   1:05p
BIGMICRO   .env        1,114 04-28-94  10:50a
VME        .env        3,812 02-22-95  11:31a
SCANTIME   .env          836 05-25-94  10:39a
SMPC       .env        1,830 08-25-94   3:08p
TIME_LL    .env        1,048 07-21-94  12:23p
        9 file(s)         18,740 bytes
                       54,296,576 bytes free

C:\MODSOFT\PROGRAMS>
```

Figure 27 Sample DOS List of Existing .ENV Files

By comparing these two lists, you can see the program PCFLDEMO has no .ENV file associated with it. Now you are ready to do steps 3 and 4.

Proceed using the new program name PCFLDEMO. Modsoft stops at the program path screen allowing you to change it, if necessary. In this example it is not, so just press the return key. Modsoft now presents the Configuration display where you also either elect to make changes or Esc. Modsoft now loads the program and displays the program Segment Status screen. As with prior functions you could make program changes or simply exit via the **Esc** key. At this point Modsoft writes the program file set to the program path including the newly created .ENV file.

You can verify the above sequence by doing another List of existing .ENV files, which will include PCFLDEMO.ENV, or execute the Modsoft Offline "Select program function" and see that PCFLDEMO is now in the list.

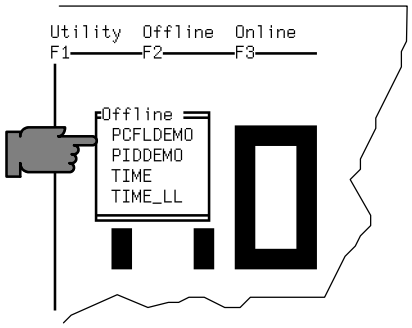


Figure 28 Updated Program Select List

1.9 Modsoft Version Upgrade

1.9.1 Modsoft 2.6

This document, 890-USE-115-00 covers Modsoft revision 2.6 which includes the development changes needed for the Momentum Magnum controller. the new BNO (Branch) module, and several new modules not in Modsoft 2.5. The following features and functions are included:

Adds the New Momentum Magnum controller.

Adds the New M1/Magnum memory sizes.

Support Magnum IR port at 19.2k baud.

Support full capability of number of points handled on I/O Net.

Support BNO Branch module on I/O Net Including appropriate I/O Map screens.

Add high end Quantum 434 and 534 PLC's and allow save user program to flash.

Support Download exec for Compact and Micro

Support Xmit builtin in Momentum

Fix numerious IPR's

1.10 Some Common Terms and Abbreviations

486 Referring to the processor in the controller being of the class 80486.

ASCII American Standard Codes for Information Interchange. In this context, a mode of Modbus communications.

ATIO Adaptable Terminal Input Output. A series of modules used as base to tophat adapters.

BUG Problem with a product, usually software.

CHS New terminology for Controller Hot Standby.

CGA Color Graphic Adapter. The earliest and least capable video monitor adapter in the IBM PC.

COMx Logical name of DOS communications port. x=1-8.

DOS Disk Operating System. In this context it means the Microsoft DOS in IBM and compatible PC's.

DROP A set of input & output slot positions that are grouped on a common backplane, bus or set of racks.

EGA Expanded capability Graphic Adapter.

IBSS Interbus-s: In M1 this is called IONET because it is not a full implementation of interbus-s

IOMAP A mapping of Input and Output points to PLC registers and discretes. Previously called Traffic Cop. An area in the controller configuration memory.

IONET Interbus-s; in M1, this is known as IOnet (no support for PCP protocol).

IR Infra Red. Magnum will have an IR port which allows wireless comms as a future enhancement.

LPTx Logical name of DOS printer device driver. x=1,2,3.

M1 A model of Momentum PLC adapter (see also tophat).

MGA Monochrome Graphics Adapter for Video Monitors.

Momentum A brand of PLC and IO module products marketed by Schneider Automation Inc.

MomentumI/O These are a series of base modules that can be physically adapted to Momentum M1 Controller.

MSL Modicon State Language.

PAGE Referring to PLC memory, 64K words.

PLC Programmable Logic Controller.

PRT Device Name of the DOS generic Printer driver.

PSR Problem Solution Request; A formal Schneider Automation system of tracking product problems and their solutions. Applies to released products of all kinds.

QIO Quantum I/O type.

RAM Random Access Memory.

RIO Remote Input Output.

RDE Reference Data Editor.

RTU Remote Terminal Unit. In this context, a mode of Modbus communications.

TOPHAT A term used to describe the IO adapters for the Momentum IO bases. M1 is one such TOPHAT adapter.

TOS Table Of Symbols.

VGA Very expanded Graphic Adapter.

VME An industry standard BUS (backplane) for computer equipment, originally from Motorola.

XREF Cross Reference. The source or destination of a PLC reference item.

Chapter 2
Modsoft Structure

□ This chapter provides you with an orientation of the displays you will see while using the Schneider Automation, Inc. Modsoft Programming panel application software.

You are also instructed regarding the common key strokes needed to move around and select items from the various editor based functions.

2.1 General User Interface

Modsoft runs with a monochrome screen as well as with a color monitor. If a graphic card is used with a monochrome screen, you must set an environment variable accordingly (Set Screen=Mono, Set Power984=RV) or use Modsoft.upf settings.

> ☞ **Note** This software does not support old versions of CGA cards; it runs, but there is “snow” on the screen.

The screen is generally partitioned into two sections. One contains the menu line, which is the first line on the screen. The rest of the screen is covered by the working area of the functional editor. Below is the primary application entry screen, this one having the entry level, or ***main menu*** on the top line.

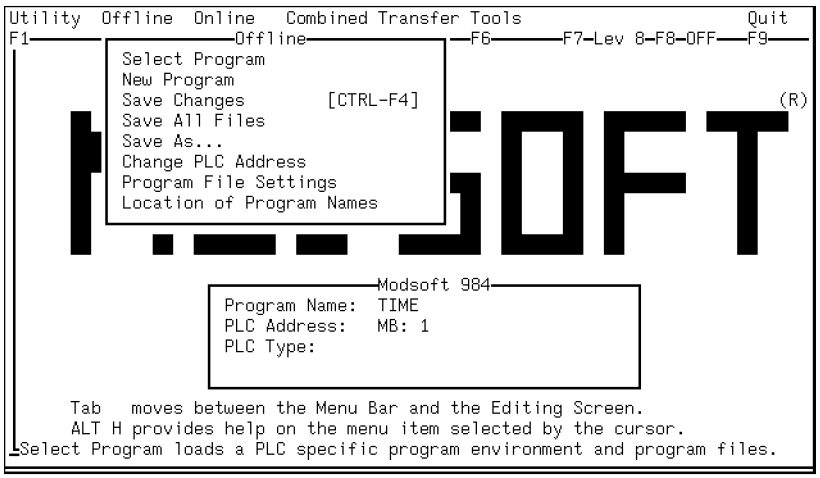


Figure 29 Initial Screen with Main Menu Bar

The menu line can be activated from any on-screen editor, at any time, by pressing the <**Tab**> key. When the menu is active, you can select functions by moving the cursor to the appropriate menu entry with Back and Forward Cursor keys as illustrated (two possible keyboard cursor movement clusters illustrated):

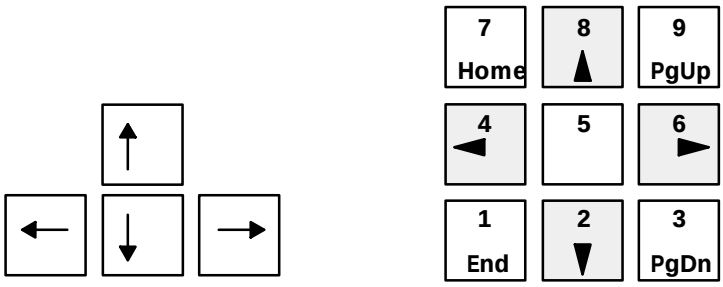


Figure 30 Menu Select Cursor MovementKeys

The active menu entry is highlighted as the cursor moves to various entries.

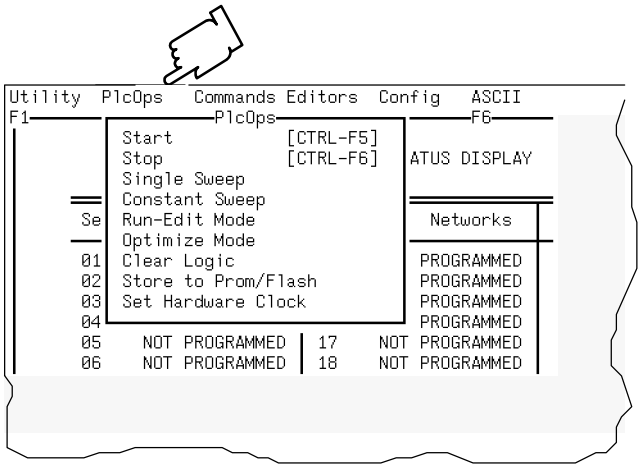


Figure 31 Menu Function Select Cursor

Assigned *pulldownmenus*open automatically, allowing cursor selection of the command function.

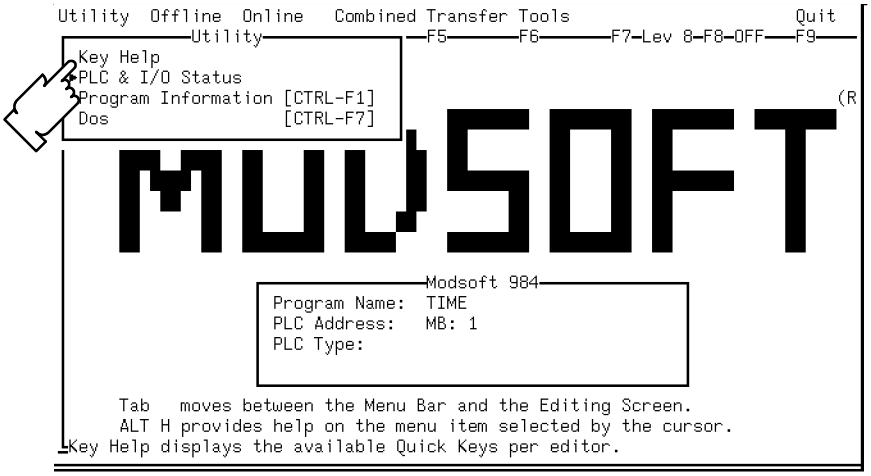


Figure 32 Pulldown Menu Cursor

Other than moving the cursor with cursor keys, there are two additional methods you can use to select functions.

- With an active menu or pulldown menu, functions can also be called by a first character selection. In the event of duplications, the second and subsequent characters determine the select movement. The following provides an example of character select functionality.

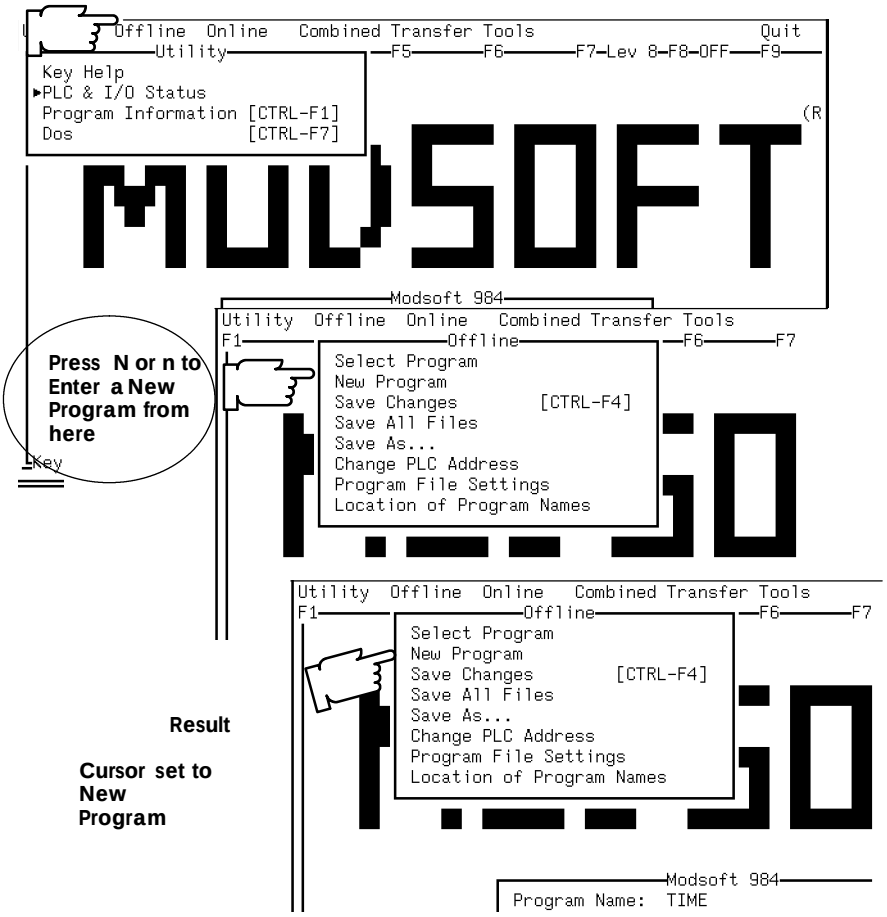


Figure 33 Examples of select by "Character"

- The function keys < **F1 - F9** > as seen on the screen under the menu entry.
  (Function keys are also effective at sub-menu levels)

The keyboard Function keys offer the same results in one less step as long as you do not mix Alphanumeric and Function keys. If you press the **<F2>** key, in the top sequence above, The Offline submenu selection appears on the menu line instead of a pulldown, then press the <**F2**> key and see the New filename entry line, which is the equivalent of the third step above after the Result is executed. The File name in this example allows you to name a program file. This point is typical of an entry point where you are beginning to develop a new program.

Using function keys presents a different menu organization and a shortening of the select statement. So that you will know how many levels away from the main menu you are there is one down arrow head displayed on the right display margin for each level to a maximum of two.

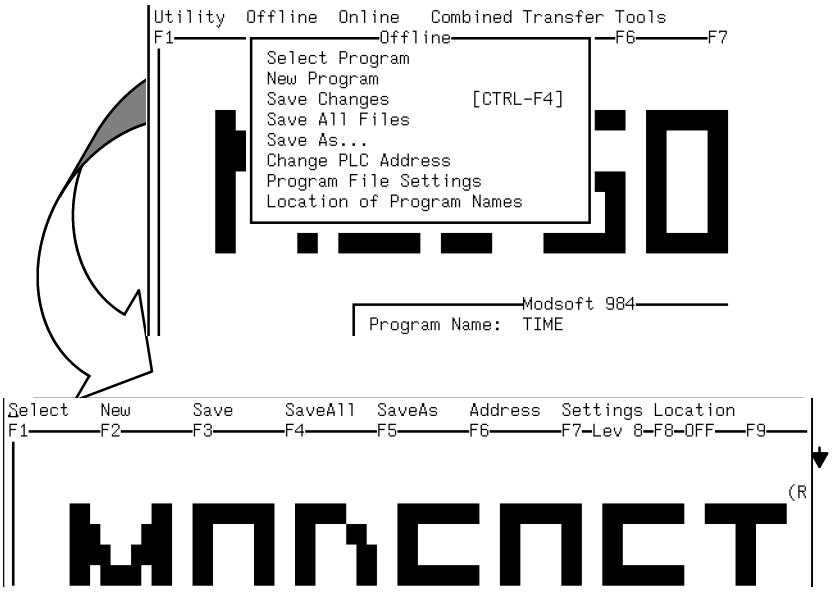


Figure 34 Example of Menu Presentation

This document may be used by beginners or occasional users, so the full cursor movement is used in examples. The more frequently you use Modsoft the more likely you are to gravitate to the use of both Function and Shorthand keys thus achieving a higher level of efficiency. The initial Program display, is a result of ***selecting*** the Filename TIME (using the above sequence) then loading the Example TIME into the panel.

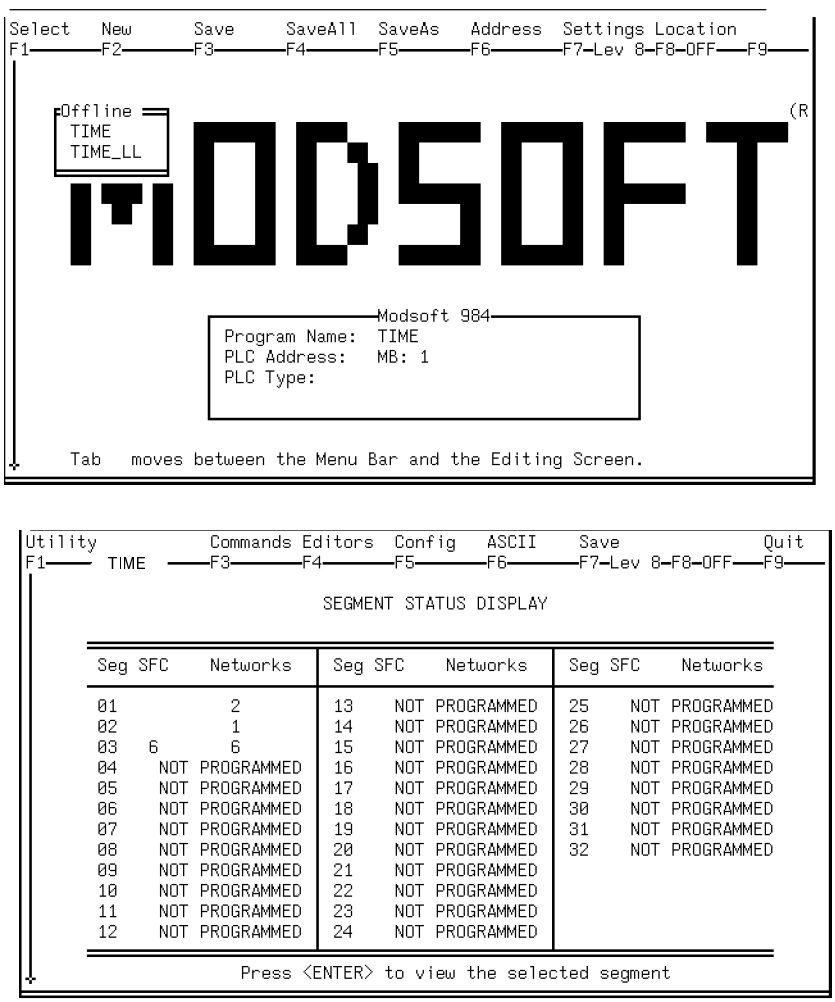


Figure 35 Example Of Time Program Select and Result

When keyed with the <**Control**> key pressed, the Function keys provide Quick access to often used features, where:

<**Control**> <**F1**> Is the Quick Key for Modsoft Information
<**Control**> <**F4**> Is the Quick key for Save
<**Control**> <**F5**> Is the Quick key for Start (Online and Combined Mode Only)
<**Control**> <**F6**> Is the Quick key for Stop (Online and Combined Mode Only)
<**Control**> <**F7**> Is the Quick key for DOS

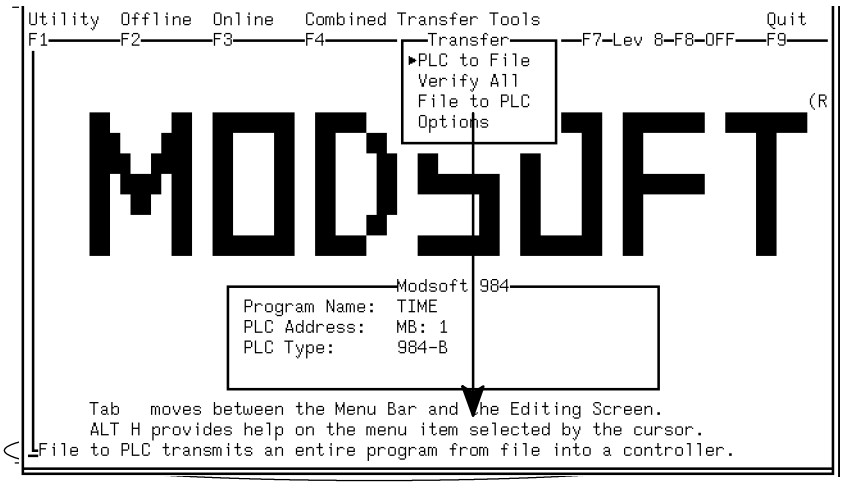


Figure 36 Example Of A Tracking Help Line Description

Additionally, for most pulldown menu entries, a selection list for Quick key selection information is assigned. When the menu is active, only cursor keys have any effect. When the editor is active, you can access special keys (e.g., cursor keys) or may directly input some text (e.g., comments). Also, for some frequently used menu entries, abbreviations are supplied.

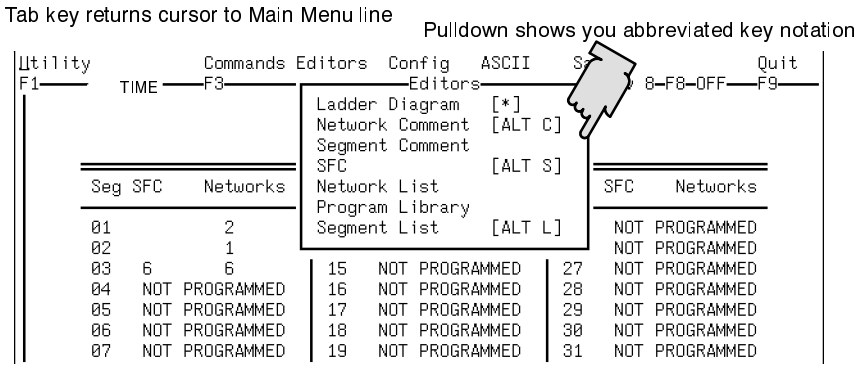


Figure 37 Frequently Used Keys and Abbreviation Example

Thus, for calling these functions, the user does not have to retrace steps to arrive at the proper menu.

The name of the project (.env), (not the .prg), is on all screens opened after selecting a project; and, it is displayed in reverse video. The name is placed in the F2 position on the top border.

Exiting the current menu can be done by pressing the <**F10**> key, selecting **"QUIT"** or pressing the <**Esc**> key. Furthermore, if one input field allows the selection of a certain set of default values, the input can also be done via a displayed selection list, and is invoked by typing <**?**> with the editor active. The selection list is shown in an extra window. If there is only one Modsoft file type available there is no list because the file is automatically posted to the filer display.

Text input on the Editor Screens is normally finished by the <**Enter**> or <**Return**> key (depending on the keyboard), but also can be done with <**Cursor Up** ↑> and <**Cursor Down** ↓> keys.

The <**Esc**> key performs the following operations:

- If an (input) operation is active, this operation is aborted and the previous field value is restored.
- If the cursor is in an edit area it is returned to the Menu line.
- If the cursor is on the menu line and there is no assigned pulldown menu, the previous menu is activated (i.e., here <**Esc**> has the same effect as the "Quit" command).
- If the cursor is on the menu line and a pulldown menu is shown, the pulldown menu is removed, and the menu stays active.
- If the menu is active and has an associated pulldown menu which also has an assigned selection list, the selection list is removed from the screen and the pulldown menu stays active.

2.1.1 Text Select

Existing text can be modified by moving the field cursor to the text field and pressing the `Insert` key. When this is done, the character cursor is activated. For every menu line, context information is provided.

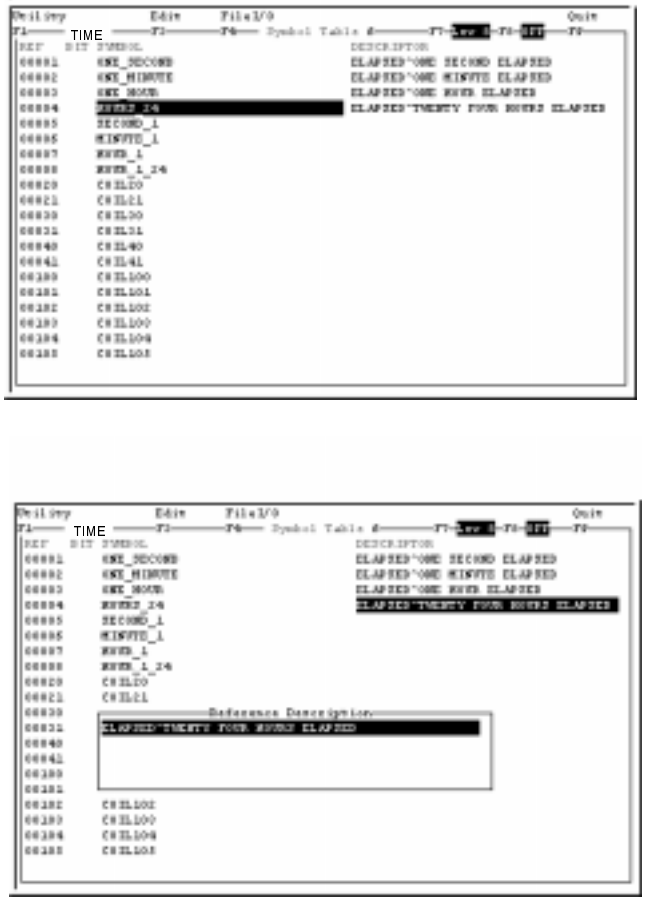


Figure 38 Example of Insert In Existing Text Field

In the following Reference Data Example, Text, data and status is made available by moving the cursor within the menu to the field entry, keying in the reference number and pressing <**Enter**>.

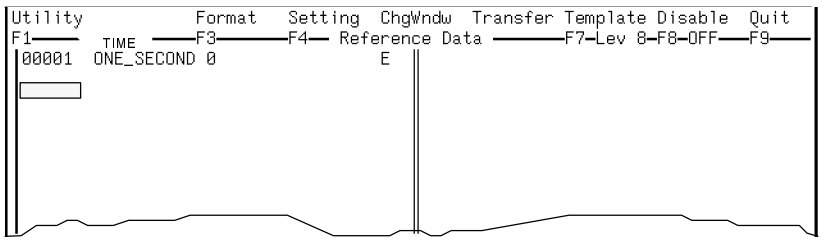


Figure 39 Example of a Reference Data Display

Detailed help information concerning individual items in the menus, can be displayed by moving the cursor on the chosen main menu entry and pressing <**Alt-H**> for Key help as illustrated. The help information is displayed in an overlaying special help window. Specific help is available at sub menu entries where for example information on Print help. Additional functionality using <**Alt-H**> can be obtained in Ladder and I/O Map editors to provide help on DX's or I/O Modules

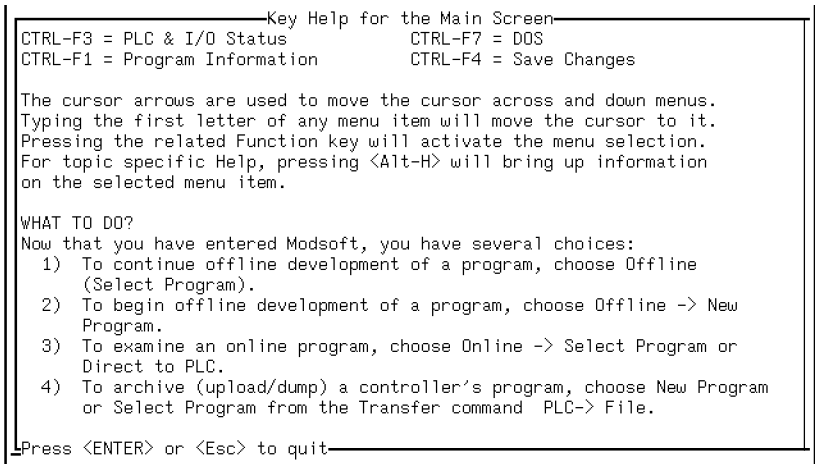


Figure 40 Example of Key Help

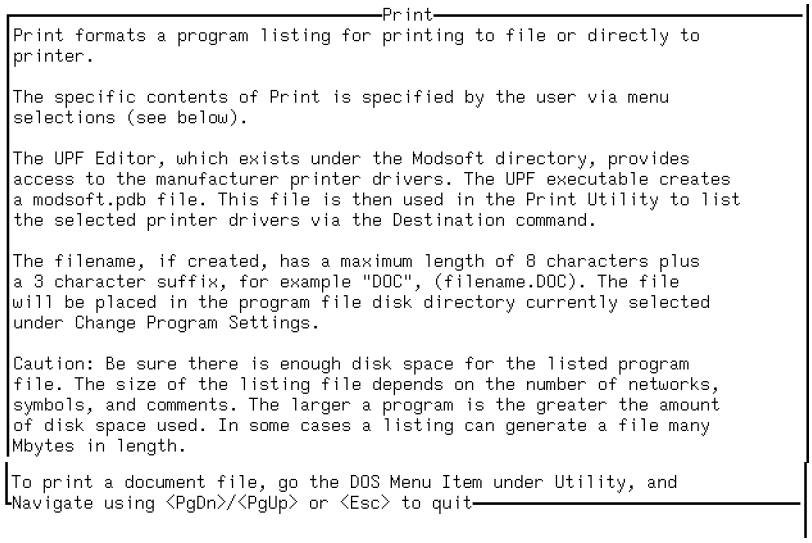


Figure 41 Example of <Alt H> Specific Help

System messages (error messages, warnings, and dialogue messages) are displayed in a separate window.

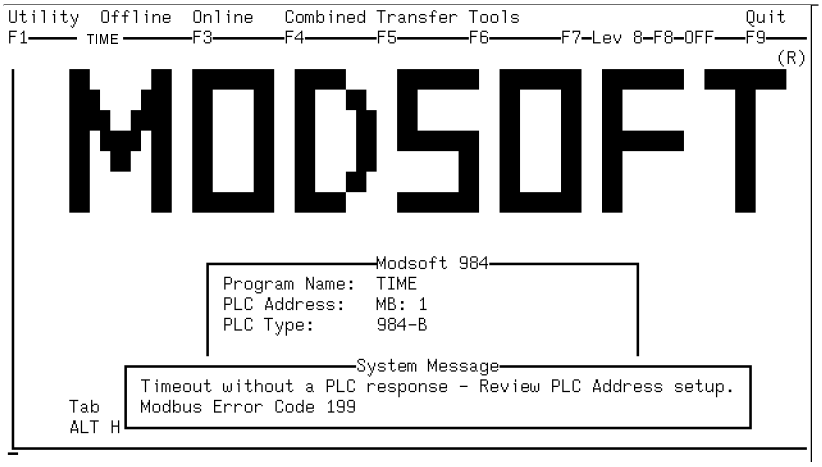


Figure 42 Example of System Message Display

2.1.1.1 Quit

"Quit" returns control to the previous menu. If you are at the Main menu entry point and press quit, you are asked to confirm the exit from the system. If you have not already saved changed files, before the system returns to DOS control, you are reminded to save the files before exiting to DOS.

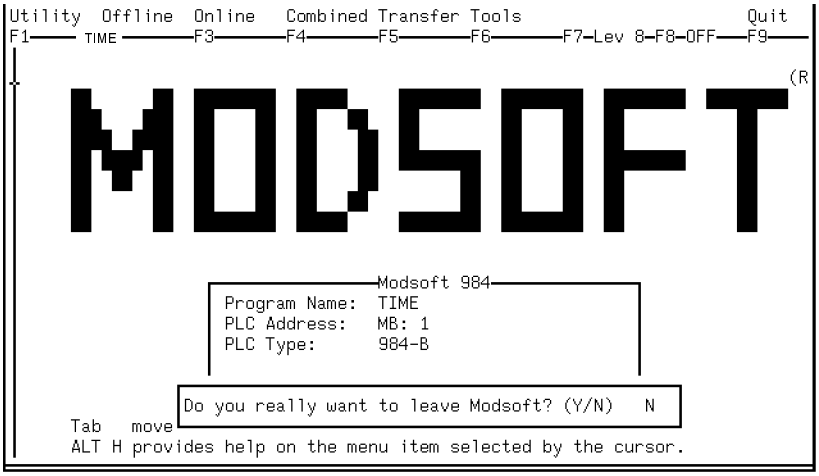


Figure 43 Main Menu "Quit" Selection

When you select the "Y' entry to signal the intent to quit the application, you are presented with a final disposition screen that requires you to respond to one of three ways to proceed. This technique insures that you reconsider the effect on the program file structure based on closing the application. For example, if you have created something new, the files in temporary locations might need to be saved to the program file directory but you may want to rename them first. Another scenario could be that some testing was done on an existing program and you may not want to save any changes. If no changes are made the exit is taken directly.

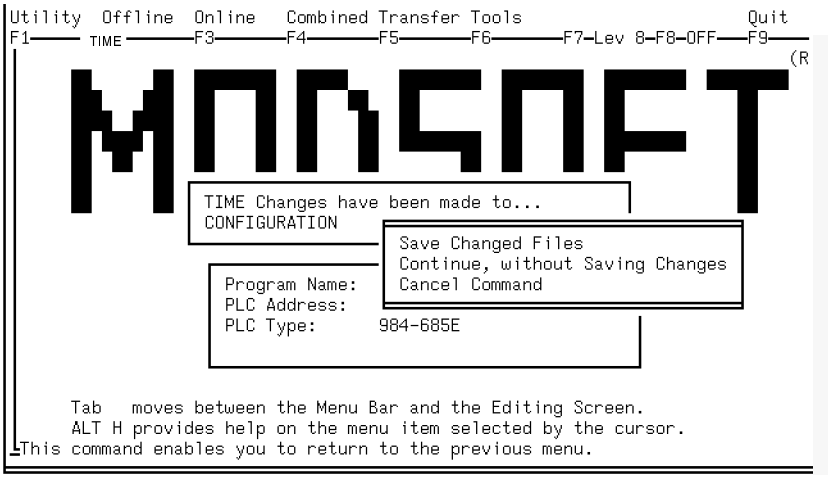


Figure 44 Modsoft Exit File Status Prompt Screen

The Save selection screen is also presented any time data can be lost if you do not take precautions. In addition to the Quit/Exit decision, the save screen is posted at Transfer from PLC (online data from the Controller). Also, a selection list is displayed when Loading, and any time one or more changed files exist or might exist because the program file was opened. The choices are:

- Save
  When selected this causes Modsoft to perform a `Save` (meaning all the PLC program related files currently in the system are saved). To continue the Save operation from the Overwrite warning you must acknowledge with a <**return**>.

- Continue
  When selected, this function proceeds to the next function without saving any changes

- Cancel
  This selection allows you to remain at the current screen. It does not save your programs and it will not continue to the next function.

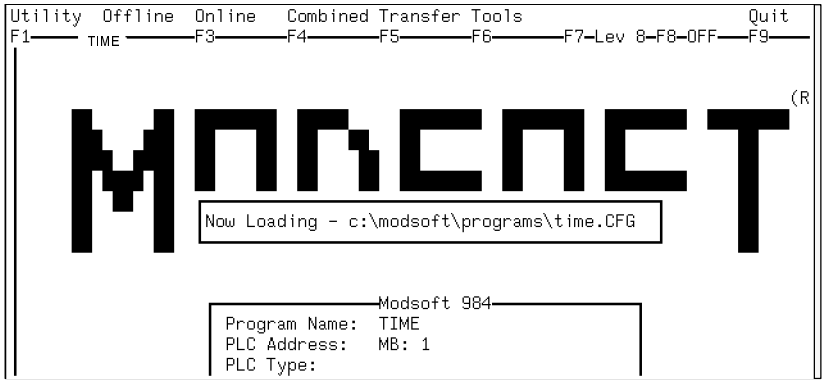


Figure 45 File Activity Indicator

For your reference, the Program name you select or create is posted to the Main menu screen. You are also advised of the Modbus Port you select to communicate with the PLC (Modbus Port Address 1 above) and if you access the PLC its type is also posted.

2.1.2 Important Key Summary

The following list summarizes the important keys and their functions:

Tab	Toggle between menu bar and editor screen
Esc	Stop entry or back out one menu.
Enter or Return	Select an entry; finish (execute) an input.
F1 - F9	Select the respective entry from the menu bar
Alt1 -9	Select menu as a pulldown
Shift F1 - F9	Get Help for respective menu entry
F10	Leave the current editor for the previous editor.
Alt-H	Help on the current entry on the menu bar.
?	Popup a selection list of entries where a number of selections are available. If one input field allows the selection of a certain set of default values, the input can also be done via a displayed selection list, and is invoked by typing <**?**> with the editor active. The selection list is shown in an extra window. If there is only one Modsoft file type available there is no list and the file is automatically posted to the file display.
PgUp/PgDn	Quick scrolling in various editors
End	Go to the bottom of a list.

When typing an entry:

HOME	Move to first character
END	Move past last character
Ctrl→	Move to beginning of next word
Ctrl←	Move to beginning of previous word
Ctrl K	Many operations that are known to consume time offer this key as an abort if you change your mind after starting the process.

☞ **Note** Since the CPU may be servicing another process, the effect of pressing the Ctrl K key may not be immediate. Keep trying.

INS	Toggle between Insert mode and Overwrite mode *Insert has an underscore cursor and overwrite has a block cursor.*
Ctrl ENTER	Conclude entry and accept the entries made in the dialog.
ESC	Conclude entry and discard any changes made.
TAB	Move to 11th, 21st, etc, field positions.
ENTER	Go to the next field.

Chapter 3
Main Menu Functions

- Chapter content is consistent with the Main Menu function selections. Some functions available from this menu are best presented as independent chapters such as the *Print* function, while others are combined with similar subject matter such as inclusion of *all* programming modes within the Ladder Logic Programming Chapter.

  The major operating modes, Offline, Online and Combined descriptions are distributed between chapters.

Utility	Offline	Online	Combined	Transfer	Tools	Quit
Key Help	Select Program	Select Program	Select Prog	PLC to File	Print	
PLC & I/O Status	New Program	Direct to PLC		Verify PLC	Macro	
Program Info	Save Changes			File to PLC		
Dos	Save All Files			Options		
Quantum Backplane	Save As...			Download Exec		
	Change PLC Address					
	Location Of Program Names					

Figure 46 Main Menu Header and Sub Functions

3.1 (Utility) Key Help

3.1.1 User Aid

The first pulldown menu selection from the "Utility" selection provides the Key Help screen. This screen, an example of which is shown below, tells you which shorthand keys are available, along with how to get more information, how to use the quick key abbreviation, and how to move the cursor. The information provided to you is dependant on each editor type.

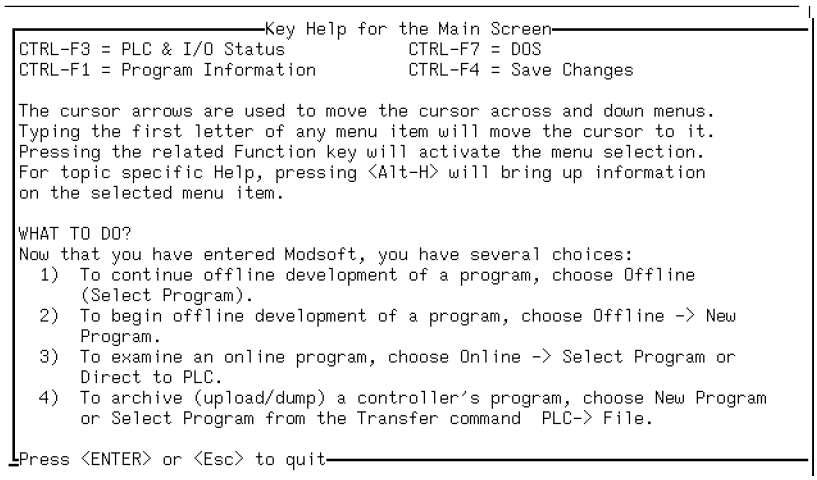


Figure 47 Key Help Selection from Main Menu Utility Function

3.2 (Utility) PLC & I/O Status

The "PLC & I/O Status" screen reads controller data. The current controller's configuration is displayed for your review. This function does not require that you be "Logged in". If the panel port cable connection is wrong or parameters are set incorrectly, a "Time-out" error (#199) or other communications error appears after the appropriate retry time has elapsed. When you invoke this command an initial status screen is returned. The following is an example of the status display.

☞ **Note:** "Logged In" indicates your login status. However, it does not indicate whether another master is logged in via a different port or path.

When you elect this function you are given a submenu from which to:

Select a Program (automatically loads the communications parameters)

or

Direct to PLC. (allows you to select specific parameters)

- The Controller Executive and Version identification codes are listed for you as is the Hardware ID and it's revision level.
- If a "STOP CODE" has been encountered on the status screen, **<Esc>** to Quit the PLC Status Screen then position the cursor at the PLC & I/O Status selection under the Utility menu and press **<Alt H>** . You will see a help screen that lists the general controller stop codes.

☞ **Note** If you are downloaded to a specific Controller, have that program logic in the panel and then you use `PLC & IO Status to select` another program, addressed to the same controller, only the orignal contents of the panel and PLC are available (that is what is running).

If However, the selected program is to a different controller the second controllers *status* is displayed *(assuming it is currently running the selected program)* for your observation. You can not interact in any way with the second controller and when you leave its PLC&IOSTATUS screen the original controller status is displayed and you can continue with panel operations on that controller (that is what is running).

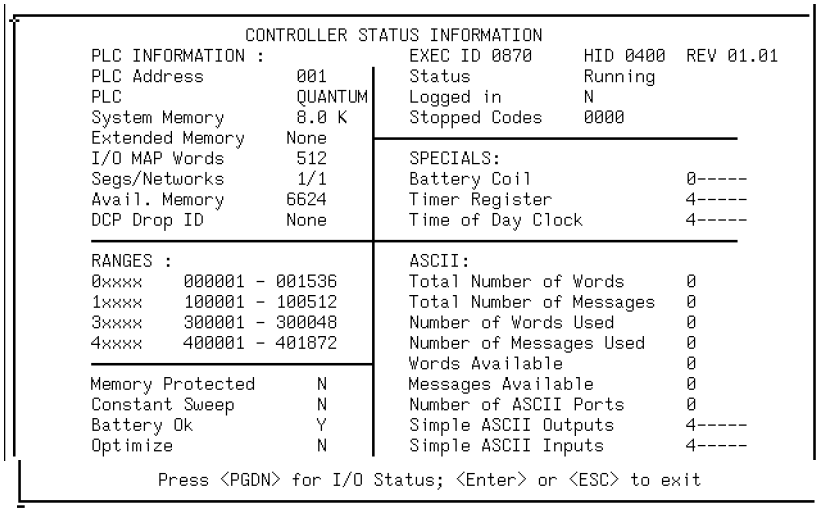


Figure 48 Example of "PLC & I/OST ATUS" Screen

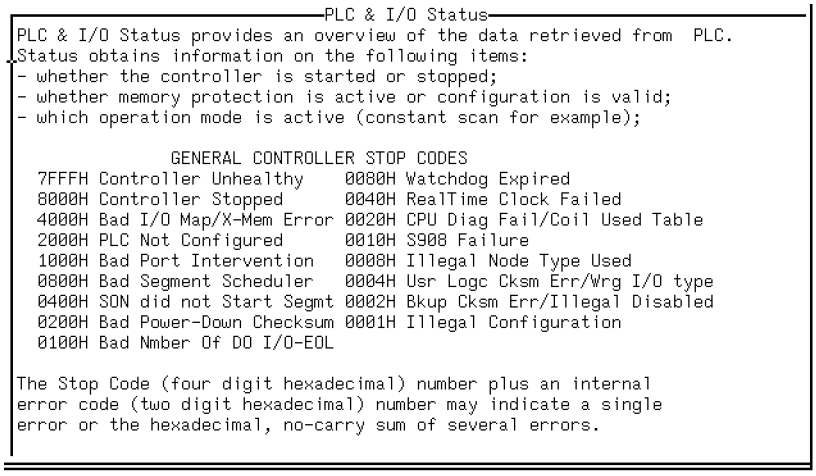


Figure 49 PLC Stop Codereference Display

Details on the I/O Drop status information extracted from the controller can be found in the Modicon Ladder Block Library User Guide 840 USE 101 00. The four character Hardware ID field is interpreted as 0100 meaning Major release (first 2 characters) 1 Minor revision level (last 2 characters) 00.

Pressing the <**PgDn**> displays the first page of Controller status table and 800 Series I/O (depending on controller type selection). The 200 Series are presented in about half as many screens.

Since there are numerous pages of information an index is available to help direct you to the information you want to observe. Due to the scrolling nature of the status table, after you page down to enter the table, if you do a page up you will wrap to the last page of the table which is the second of 2 index pages. The index Topics and their page assignments are:

STATUS ITEM	PAGE	STATUS ITEM	PAGE
PLC State	1	Hot Standby	2
Local System	2	Remote System	2
Single Sweep	3	I/O Error Code	4
PLC Error Status	5	NBR Segments	6
End Logic Pointer	6	RIO Timeout	7
Message Errors	7	Run/Load/Debug	7
IO Health Drops 1-3	8	. . . .	. . .
IO Health Drops 31 - 32	18	QTM Start Errors	19-20
Cable A Errors	21	Cable B Errors	22
Global Coms	23	Remote IO Drop 1	24
Remote IO Drop 2	25	. . . .	. . .
Remote IO Drop 32	55	DIO PLC	56-59
DIO Option 1	60-63	DIO Option 2	64-67

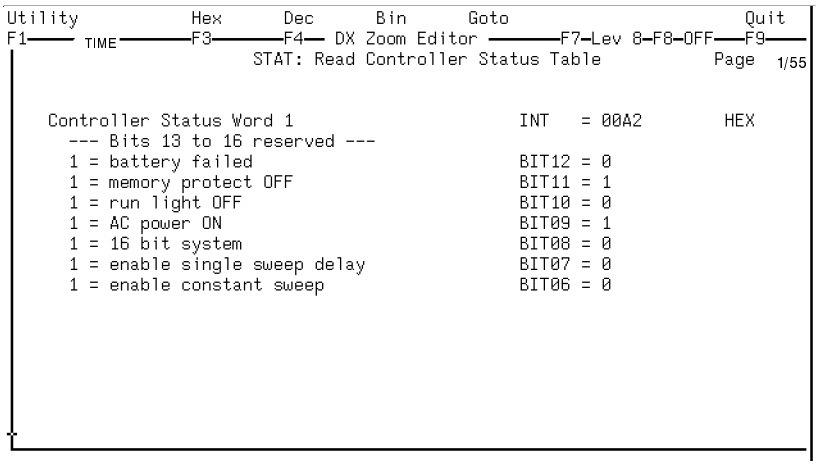


Figure 50 Page One of I/O Drop Status Table Display for Example

3.2.1 Status with a Quantum Selected

When you have configured a Quantum Controller, additional pages of status are added for health information. The initial status screen indicates that health is continued on page 56:

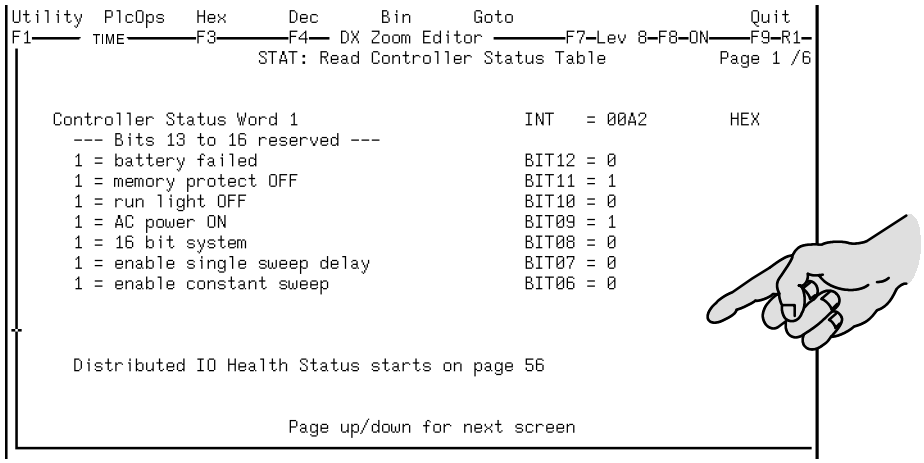


Figure 51 First page of I/O status with Quantum selected

3.2.1.1 DIO Health Status Screens

Modsoft has a display of module health status equivalent to the S908 ”STAT” display.

In general, the health bits have the same meanings as RIO drops, i.e. the module is healthy or not. The screen layout will be as close as possible to health bits display for s908. The significant change is in the use of memory pages A0 -> FF for storage of the health bits. You do not need to be aware that the data is in pages A0 to FF. An example of the page 56 layout follows:

☞ **Note:** The above PLC Status including Module health is also available for the M1 controller in Modsoft version 2.5 and above.

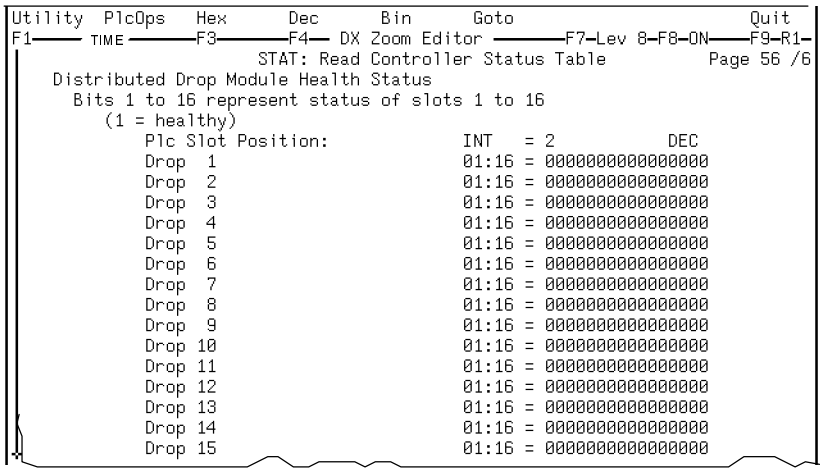


Figure 52 Beginning Page Of D I/O Health

HEAD 1: Quantum DIO Module Health Status

Bits 1 to 16 (left to right) are status of slots 1 to 16 (1=healthy)

Note: this screen info is not included in the lister, since it is strictly “on-line” information.

3.3 (Utility) Program Information

3.3.1 Modsoft File State

The "Program Information" screen provides information on the offline environment. This screen displays the controller type, configuration range, and file information, and includes revision and date information. The next screen displayed, after acknowledgement of the error, displays the last edit session pathname.

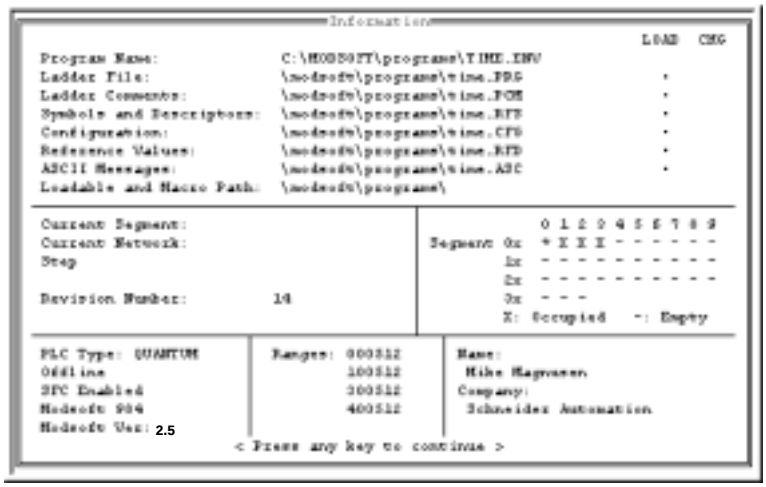


Figure 53 Information Screen Example

File information is provided in the form of prompts to indicate the files in the set that are loaded and a similar indicator to identify any files that have changes pending.

The revision number is part of your document control function and is incremented each time a .PRG file (Modsoft Ladder Logic Program) name is saved.

In OnlineDirect, the revision is always 0.

3.4 (Utility) Dos

3.4.1 Application Exit

With the "DOS" selection you are in a DOS shell and can issue DOS commands, such as copy or rename. This function can be executed only if there is enough memory available. When executing "DOS", never attempt to access files that are currently used by Modsoft; this may have unpredictable results. To return to Modsoft, type **"exit".** The Login status is passed to MODUTIL.BAT (DOS Batch file) so that you may use it to customize your Batch file to communicate with the PLC Modsoft is "Attached" to (Online).

When you select the DOS option you can let the PC default to the Prompt level such as C:\MODSOFT> or ***you*** can create a batch file named ***MODUTIL.BAT.***
You can thus customize a DOS command sequence to your particular needs. If the Modutil.bat file exists in the Modsoft directory, it is executed when DOS is selected from the Modsoft Utility menu and when the execution is complete you are returned to Modsoft.

3.5 (Utility) Quantum Backplane

This functionality displays EXECutive information for all modules in the backplane **when Quantum is the configured Controller**. This function is activated from the `Utility` menu bar of the main menu but is not functional when the controller is running.

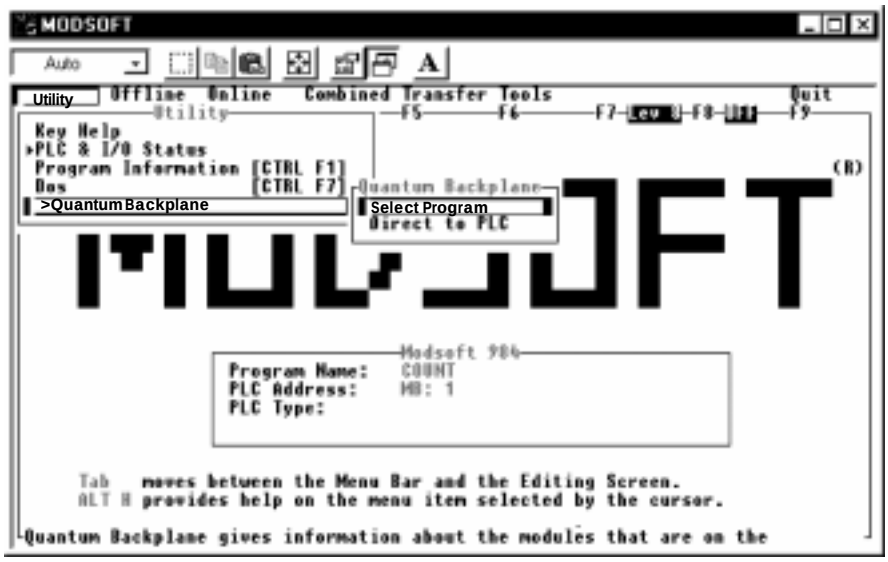


Figure 54 Utility Menu with Quantum Backplane Selected

Selection of the `Quantum Backplane` Information item from the `Utility` pulldown is followed by a request for the identity of the Controller. e.g.

Select Program or
Direct to PLC

Selecting `Select Program` or `Direct to PLC` performs exactly as the PLC & I/O status function. When you have successfully attached to the Controller, the INFOrmation screen pops up. The information contains at least the following:

Slot	Module ID	Hardware ID	EXEC ID	EXEC rev	Kernel rev
01	(0x0108)	0x0405	0x871	01.00	02.00
05	(0x010C)	0x0300	na	01.05	01.00
07	(0x001B)	0x0101	na	02.44	02.01
08	(0x010A)	0x0201	na	01.00	01.01

If a module does not have an EXECutive ID or equivalent, this data item is empty.

If the target Controller is not a Quantum, a message is displayed:

```
No Backplane Info available for non-Quantum Controllers.
```

The definition of the backplane is = rack 1.

Note The conditions necessary to access a unit in a given slot is dependent on the specifications of the unit. e.g., some hardware may not allow buss access while the interrogated unit is running.

3.6 (Offline) Select Program

Offline is the major program development applications mode. When selected from the main menu you are immediately provided with a pulldown menu that provides all the file manipulation support you need to create new Ladder Logic or load previously designed logic.

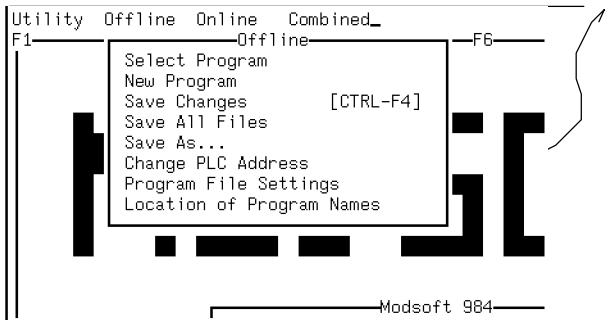


Figure 55 Offline Program Pulldown Menu

- The ability to load the panel, with your program set in a minimum of keystrokes, depends on how you set your Location of Program Names path

  Select Program EXAMPLE

  The initial setting of the Location of Program Names is \modsoft \programs

  The distribution examples “Time” and “Time_II” are in \modsoft\programs
  and the entire file set is selected with the single name entry in the program select list.

3.6.1 Use existing Program

Pressing the <↵> key, with the cursor in the first field of the pulldown, presents a displayed list of previously created programs. Your initialized Modsoft application posts the default entries that can be select using the cursor and enter keys.

☞ **Note** It is important to have a separate programs directory (folder) for 32 bit PLC's (e.g. the 424). This separation will help you segregate the User loadables between 16 and 32 bit types.

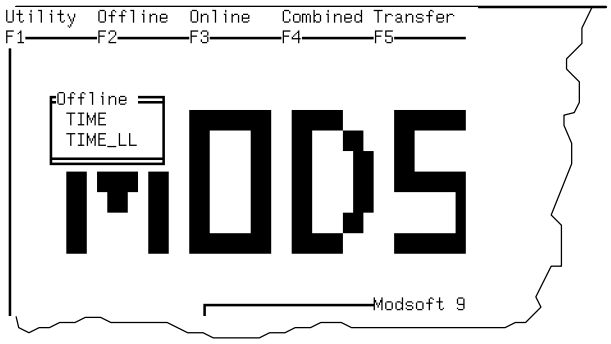


Figure 56 Example of Program File Selection Menu

This Main menu Offline function with the additional pulldown submenu allows you to Create or Load the Application program file name by selecting "Select Program" or "New Program" After making the selection, the file name you picked is referenced on the Function key line of all relevant displays.

☞ **Note** If there are approximately 116 program names or more in your projects sub-directory, the screen cannot show them all at the same time.You will note that the border for the projects selection list will contain

--<<----- project ----->>--

to indicate that you may find more names in the list by pressing either the right arrow cursor key or the left arrow cursor key.The maximum number of projects that can be shown in this expanded window approach is 252. It is recommended that you do not have so many projects in a single projects directory that even this expediency is necessary. Group your projects by geographical or functional categories to keep the number below 116 in a sub-directory.

You can create a new file (in New Program) or name or accept a program (Select Program) selection currently posted (if any) by cursor or scroll position and pressing the <↵> key. Do not give your program a DOS reserved word for a file name.

To remove a name from the Displayed list use the Utility DOS exit and delete the NAME.ENV file from the \Modsoft\Programs\ directory (or where the "Location of Program Name's has been set). Your programs remain in the \Modsoft\Programs\ directory unless you explicitly delete the entire program set..

☞ **Note** If a .ENV file (Environment parameters for a Modsoft program) is deleted by accident, you can recreate it by selecting New Program and assigning the same file set name that you want recreated. Save the new name and the .env is re-established. The program files are not changed, only the .Env file is created.

3.6.2 Selection Default

After the initial Install startup, whenever Modsoft is re-started, Modsoft reads a file to obtain its setup information that represents the program environment the last time the application was Quit. The user can specify a particular name for the startup file on the DOS command line when starting Modsoft. You might type:

C:> Modsoft Mysetup

Where “Mysetup” was previously created using the New Program function. If you do not specify a named setup file, Modsoft uses the the system copy that is always current (or at the initial default, which is Modsoft.Env).

3.7 (Offline) New Program

This option sets up the directory path, file name, and controller specific information. In the sequence that follows, you will begin to use the cursor placement and menu select keys defined in the last chapter. Some of the displays that you will see, specifically the configuration screen, are given to familiarize you with the screen. The details of all the Configuration settings are provided in Chapter 4.

The process of creating the environment in which your Logic program is to function, requires a number of user defined parameters in addition to the new program files. The task is now largely automated. The steps you will go through to create an offline program are:

Step 1 Assign a NAME to the Program.

Step 2 Assign a directory where the Program will be stored.

Step 3 Assign the communications protocol for the PLC and the address.

Step 4 You can create or alter the configuration parameters if desired.

Step 5 You can use the Ladder editor to create logic if you want to.

Step 6 You can return to the main menu and save the program and its environment.

The first two steps in development of a New program are demonstrated in the next illustration. Once you have assigned the filename, you are prompted to enter the program path. The drive defaults to your current hard drive and a path of:

\Modsoft\Programs\

☞ **Note** It is important to have a separate programs directory (folder) for 32 bit PLC's (e.g. the 424). This separation will help you segregate the User loadables between 16 and 32 bit types.

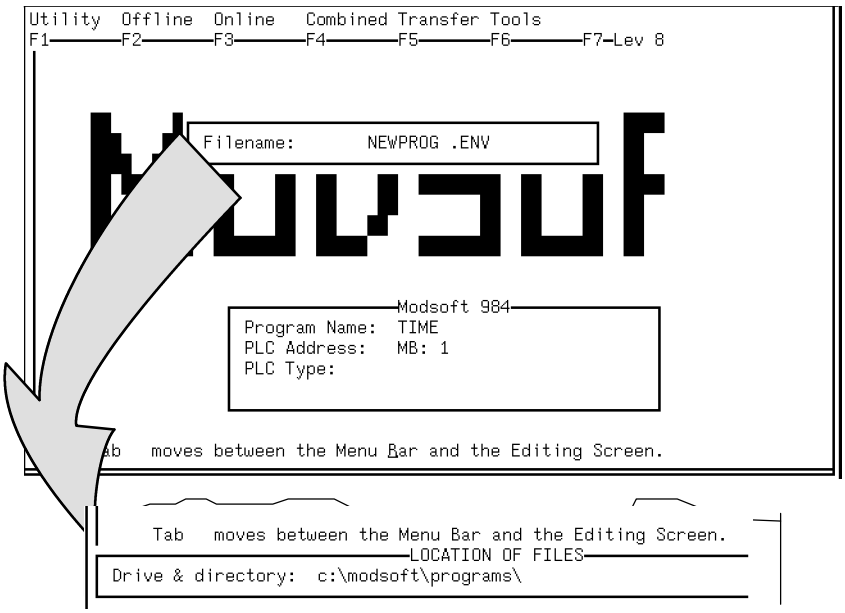


Figure 57 New Program Selection from Main Menu

☞ **Note** If you have installed Modsoft to another Drive or name, you must use the Modsoft.upf parameters to set the files properly i.e.
Names directory
Files. x2, x2 . TMP984. x1, x2, . RT984 x1, x2 etc..

After assigning the path, you are presented with the communication parameters screen. This screen has the initial default settings. You can return to these settings by resetting the Defaults. For each of the parameters, you are allowed to select different settings by selecting from display lists. If you wish to keep the default you can simply press the <**Return**> key to advance through the fields or press the <**End**> key while on the first field.

(Offline) New Program

Optional Parameters are:

Address = 1 - 247
Protocol = Modbus I
Modbus Plus
Modbus II
TCPIP
Modbus Map 3.0
Decnet
Default *

Mode = ASCII (7 data bits) RTU (8 data bits)
Parity = NONE EVEN, or ODD
Stop Bits = 1 or 2

Baud = 50 75 110 134
150 300 600 1200
1800 2000 2400 3600
4800 7200 9600 19200

Device COM1 through COM8

*** Default settings are:**
Address = 1
Protocol = Modbus I
Mode = RTU 8 data bits
Parity = Even
Stop Bits = 1
Baud Rate = 9600
Device = COM1

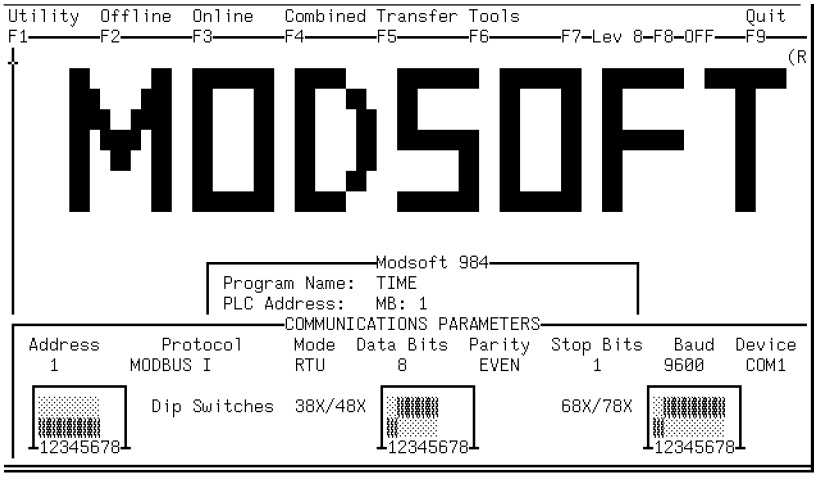


Figure 58 Communications Parameters

☞ **Note** DIP switch settings are graphically illustrated to correspond to the parameters you are setting in panel software. Your PLC may or may not have a DIP switch to establish communications addressing. Consult your PLC Installation manual for settings. (E model controllers are not represented)

When you complete your review of this screen, the selected parameters are stored and you are presented with the Configuration Overview screen. This screen provides a summary of the operating environment defaults set for the controller and is fully detailed in the next chapter. The full range of Communications Options is documented under the Change PLC Address Menu item. If you are following this sequence with an executing Modsoft application, you will see the initial prompt is at the I/O Map menu item. You can change the PLC Type, if it does not match the one you are writing the program for, by moving to Overview The type selection available is displayed for you (see Chapter 4 For details).

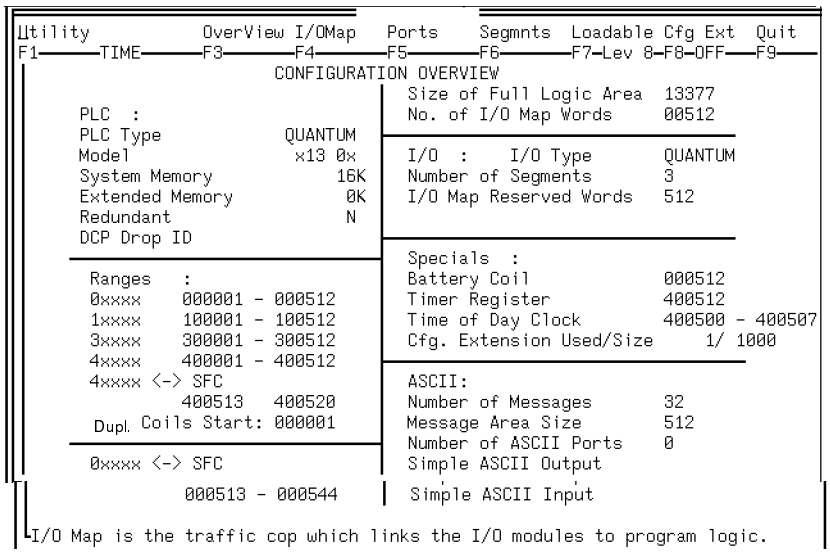


Figure 59 Example of Configuration Overview

When you exit the overview screen, the Segment Status display is initialized. The cursor is located on 01, when you press <**Return**>, you are given the prompt to open a segment for network development (if the segment has not already been used. If you want to the opportunity to continue with your first network type (Y) or end the sequence type (N.)

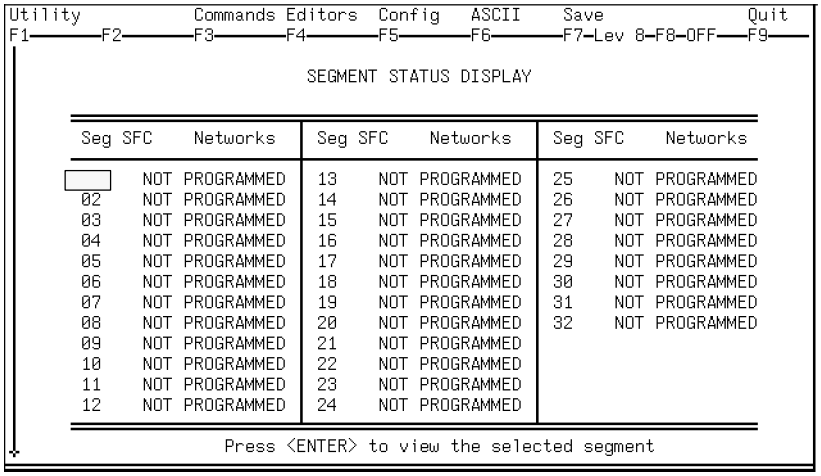


Figure 60 Direct Entry to Programming

3.7.1 Initial User

If you are an initial user of either this Panel software or Modicon controllers (or both) you can use the limited experience gained in the above sequence to bring a new controller out of the “not configured state”. You can use either the Time Program, supplied with the software distribution, and the Select Program feature or the New Program steps above. After verifying that the controller type is proper, go to the Main menu Transfer function and select File to PLC. Notice the green communications active lamp blinking while the transfer occurs. You will be prompted to Start the Controller when the transfer completes and when doing so, the RUN light on the PLC is on.

(Offline) New Program

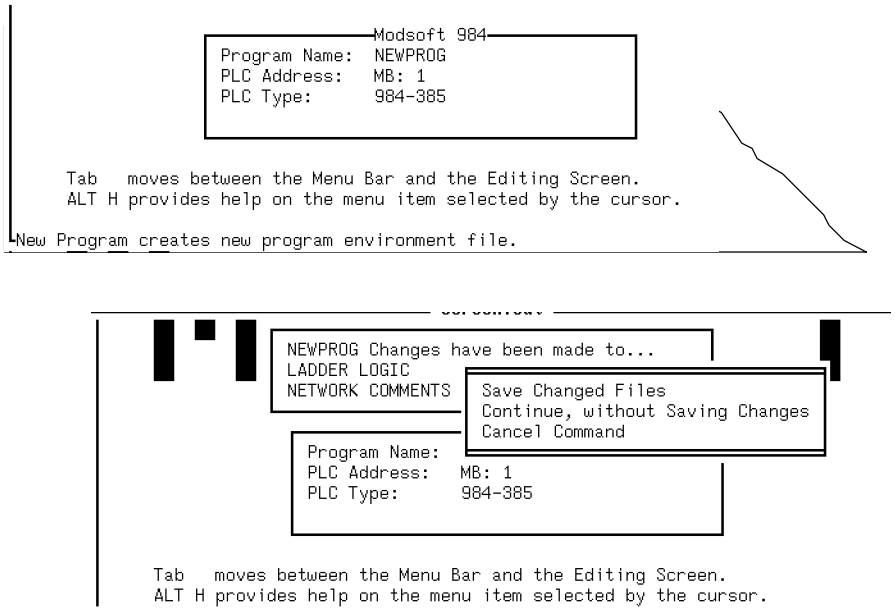


Figure 61 NewProg Posted to Program Name and .ENV

The program name, PLC Address and controller type you configured are visible on the main menu screen for you to reference. If you make any changes to any of the program files and begin to work on another file name you will be warned and advised of the action to take as illustrated.

3.8 (Offline) Save Changes

This Offline selectable function allows you to SaveChangesto files. The changes are automatically saved to the current path settings. If no changes have been made the display looks like:

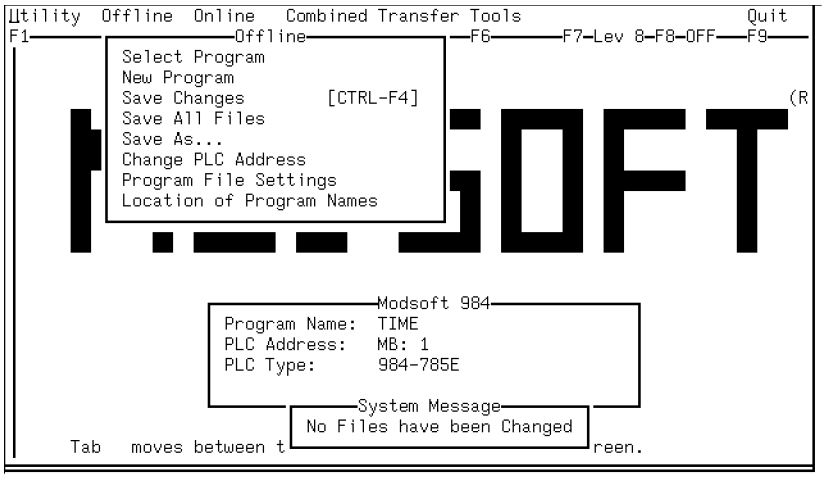


Figure 62 Main menu Offline Save File ChangeDisplay

☞ **Note** ONLY files changed during the current edit session are saved. Remember you can check the identity of the changed files by looking at ProgramInformationunder the UtilityMenu,

When saving a program, the actual revision number is written to the program header, and the internal revision number is incremented. The actual date is taken from DOS. These are available on the "Utility" Modsoft Information screen.

3.9 (Offline) Save All Files

SaveAll Files either saves the files you have just created or prompts you with the six existing base file types (See SaveAs in the next paragraph for extensions) that make up a complete controller program and asks if you want to overwrite them.

A **Y**es answer initiates a file save process and progress display as illustrated in the lower display below. A **N**o simply returns to the Offline pulldown where you left off.

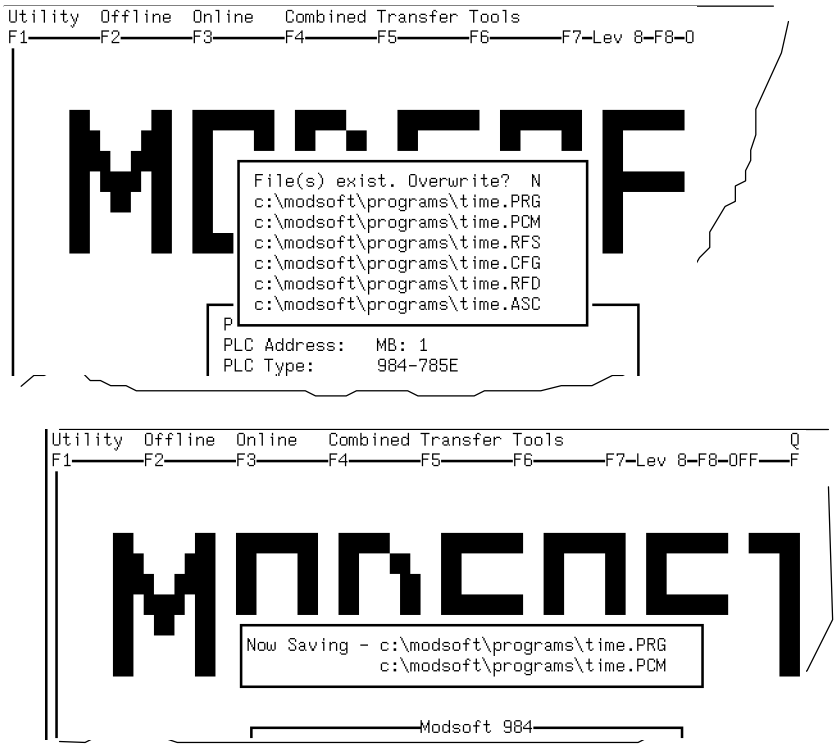


Figure 63 Offline Save All Displays

3.10 (Offline) Save As

The SaveAs option performs all the operations connected with saving a program under a different name or in a different directory or disk.

When a program is developed using the Modsoft applications system, information relating to all aspects of the data is collected in the .env file set defined in Appendix E.

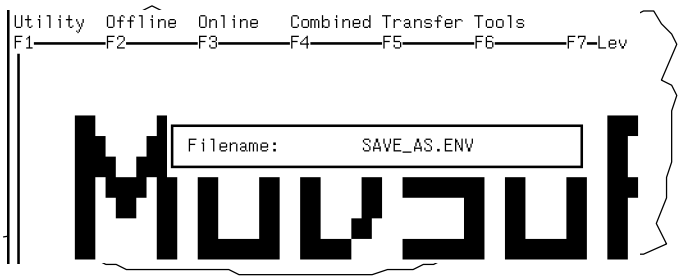


Figure 64 Offline Save AS File Name

After specifying the file name you are prompted to enter the Location of Files or accept the default. Your selection will be the path the application looks at to find the environment file for the file name you specified.

Next you are given the opportunity to change the communications parameters then the file set is saved and you can see that activity in a window on the display.

The final stage in the SaveAs process is the posting of the New program name is displayed,

(Offline) Save As

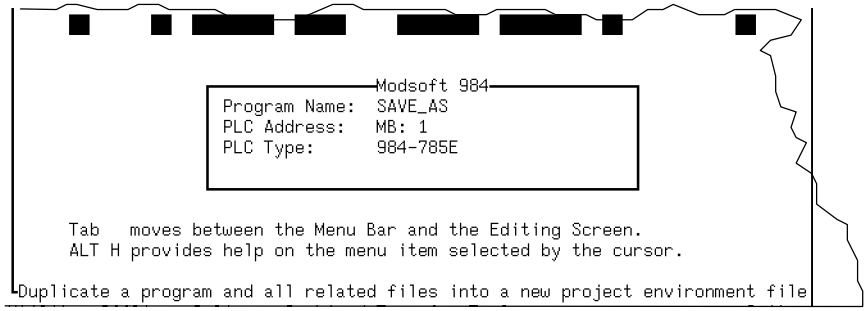


Figure 65 Window To display ChangedFile Name

The three fields illustrated above define the:

1) .ENV filename
2) Current communications setup
3) PLC Type

3.11 (Offline) ChangePLC Address

3.11.1 Communications Setup

This service provides you with the facility to set and modify controller communications addressing for network applications and Serial I/O communications. The illustration below indicates the communication parameter selection data entry fields.

You have seven different protocol choices as illustrated. The default choice sets communication parameters to Modbus I default values.

Modsoft stores the communications parameters into the program file. Thus, if you will be communicating to the same (addressed) controller as in your last session you need not access the panel port screen before going online.

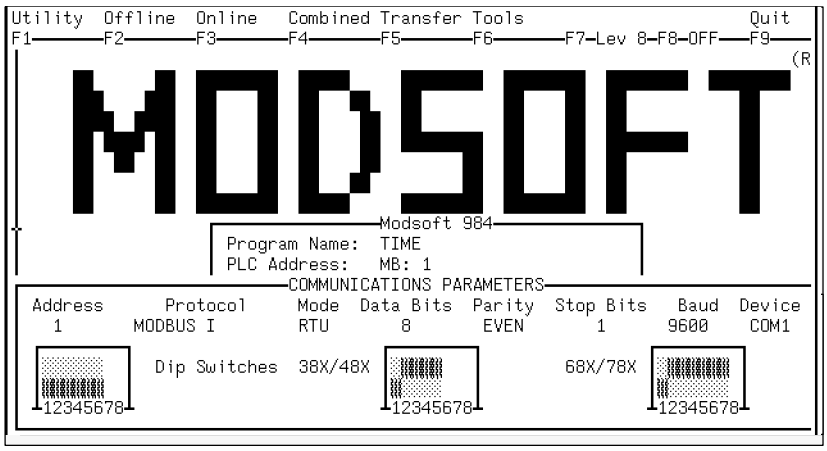


Figure 66 Communications Parameters Display

☞ **Note** Communications disciplines that provide networking also require knowledge of the addressing requirements for proper routing of panel data *and any additional hardware requirements*. See your network administrator for proper Modsoft input values. For example; SA75 and SA85 users must install the card driver ***before*** using Modsoft. The SA75 is IBM-AT based card for Modbus II and the SA85 is an XT/AT card is for Modbus Plus.

3.11.1.1 Address

This field allows you to address from 1 to 247 different PLC's

3.11.1.2 Modbus I

The top field selects Modbus I. The parameters allowed for this protocol have been documented in earlier sections.

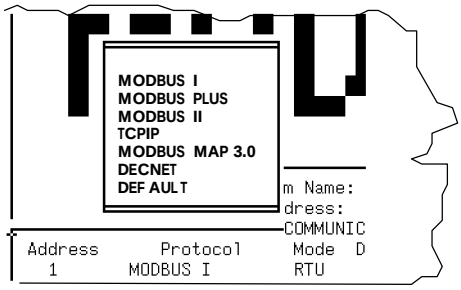


Figure 67 Protocol Choices

☞ **Note** Once a communication mode is selected, ALL the fields must be completed. You can press the <**End**> key to avoid pressing multiple return keys. The <**Esc**> key aborts the selection and returns to the original parameters.

Some of the selected modes require you to provide the computer serial port and the controller address (1 - 247). Then the serial port parameters of baud rate, parity, stop bits, and data bits must be selected.

☞ **Note** In a Momentum Configuration, the M1 can have ***only*** One Stop bit but the Magnum can have 1 or 2 stop bits.

3.11.1.3 Modbus Plus

If Modbus PLUS communication is selected, you must select the controller address, the routing path, and the LAN adapter.

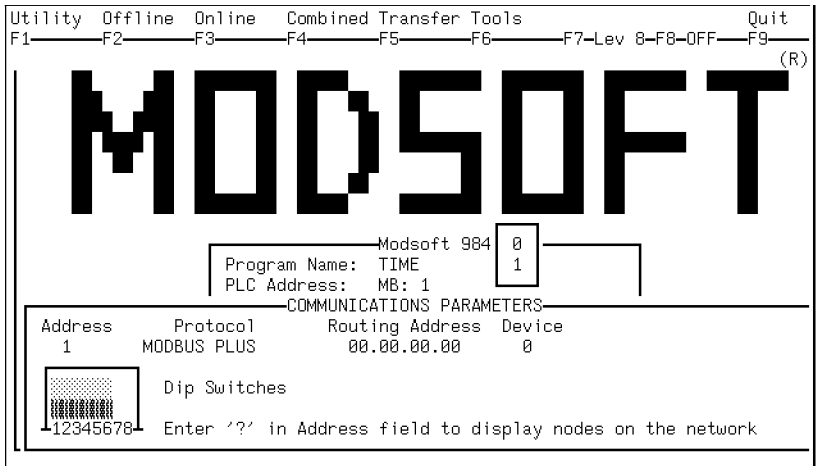


Figure 68 Modbus Plus Selection Screen

For a technical description of Modbus Plus refer to the Modicon "MODBUS PLUS Network Planning and Installation Guide" 890 USE 100 2.0. A more detailed example of routing is provided in the `Transfer - Download Exec` paragraph later in this chapter.

The Address field defines the destination device. Routing parameters are only needed if you are passing through an intermediary device such as a bridge.

If your panel has an internal Modbus Plus adapter it can act as a Host and if you select the `Online` Mode and put a **?** in the Communications Parameters *ADDRESS* field the Unit addresses of all other devices on the network is displayed. The next figure illustrates a sample network where the panel Host address is 31 (the asterisk indicates it can not be logged into). The address 1 and 2 PLC devices can be logged into from this panel.

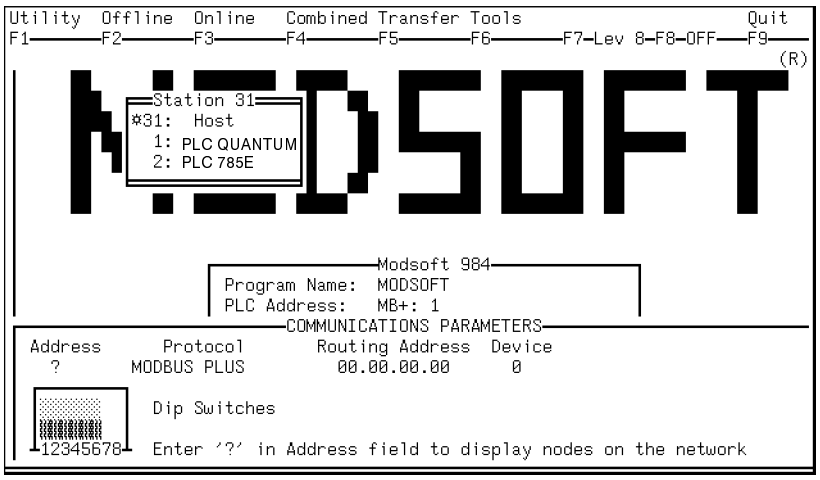


Figure 69 Example of MB+ Node Summary Display

3.11.1.4 Modbus II

Modbus II is similar to Modbus Plus in what the set up requirements are. The following prompts are displayed:

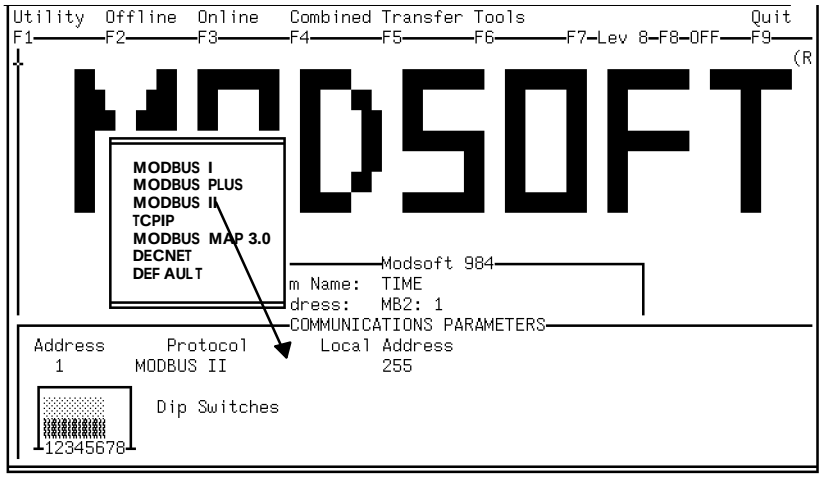


Figure 70 Modbus II Set Up Screen

After the PLC address is entered, you are prompted for the Local station ID, which has a range of 1 to 255.

3.11.1.5 TCPIP

This is the Tele Communications Protocol - Internet Protocol. Your Communications Parameters include a Protocol Identification fixed field and a prompt for the 3 variable fields:

Dest_Port Dest_Index TCPIP Address

Figure 71 TCPIP selected

- Dest_Port (NOEx 11 Default = 502)
- Dest_Index (2-225)
- TCP/IP Address

After adding TCP/IP to your configuration, when Modsoft is re-started the entry banner display will list:

Program Name: Modsoft
PLC Address; TCP/IP: xxx.xx.xxx.xxx

3.11.1.6 MAP 3.0

MAP is the acronym of Manufacturing Automation Protocol. This communications type is the result of standards definition within this specific industry. When you are using your panel (which must include the LP25 Computrol board) in this environment select the MAP 3.0 menu entry. The following screen is displayed:

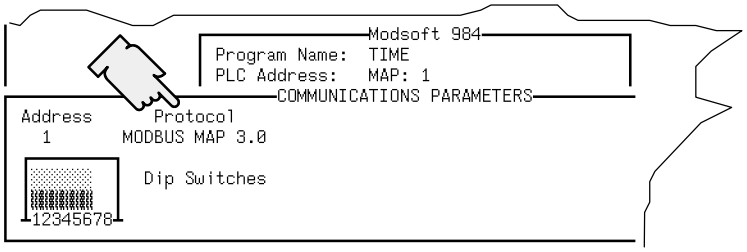


Figure 72 MAP 3.0 Protocol screen

As a pre-condition to entering the proper PLC address value, your network administrator must have created a config.adr file and included it in the Modsoft directory i.e., there is a C:\MODSOFT\CONFIG.ADR file in the format:

xxx yyyyyyyyyyyy where:

“x” is from 1 to 255 and “y” is a 12 digit MAC address established by your system administrator who also has the responsibility for installing the proper hardware and software in the host PC on which you are operating Modsoft. The MAC address is the address of the S980/984 you wish to communicate with.

3.11.1.7 Decnet Requirements

This Modsoft functionality is provided for those installations that use the Modicon Bridge Gateway Networking product with Decnet ® software. If this feature can be used, your Network administrator will provide you with the required addressing and setup information..

The next illustration provides a systems level overview of the various hardware and software elements required to use Modsoft across the Ethernet network. As illustrated, the programming panel you use requires an internal hardware communications interface and Decnet software. The Decnet software is a part of the Bridge Gateway product SW-EMBP-000. The address of the MMI hardware is determined by the network administrator and is part of the Decnet set up (see Bridge / Gateway product documentation).

The Bridge Gateway consists of the hardware, Ethernet interface and Modicon SA85 with Bridge Gateway software. The “1.93” is the node address and this number is entered via Modsoft.

Another set of address values, entered in Modsoft defines, the Modbus Plus routing between the SA85 and NW-BP85-000 Bridge (030 in this example). and the PLC address on the distribution side of the Bridge (016 in this example).

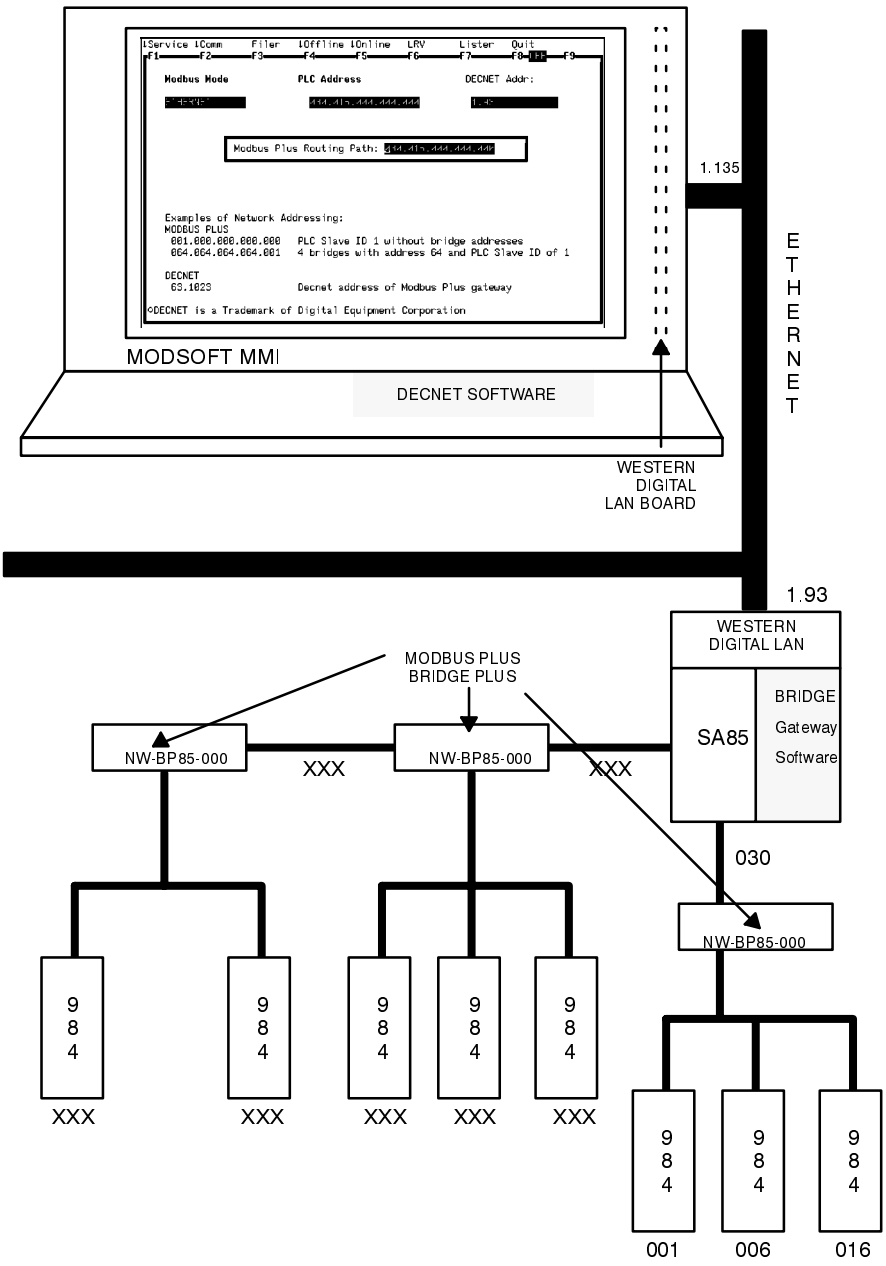


Figure 73 Decnet Overview

3.11.1.8 Decnet

If you select this network communication technique the display prompts first for the MODBUS PLUS routing path then the cursor advances to enable the Network parameters to be entered.

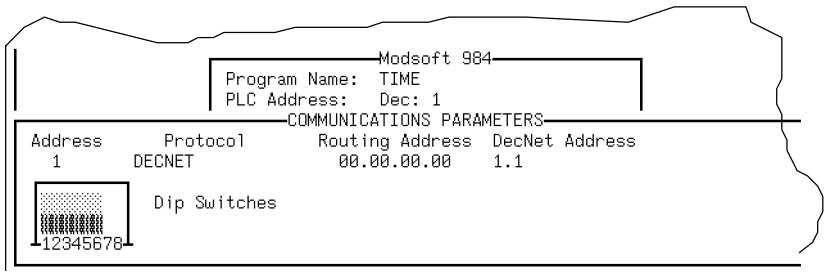


Figure 74 Decnetselected

3.11.2 Default

The values illustrated are the same as the start-up values thus giving you a known set of parameters to reset the communications mode to. The default values are specifically chosen to match new manufacture Schneider Automation, Inc. controller initialized parameters.

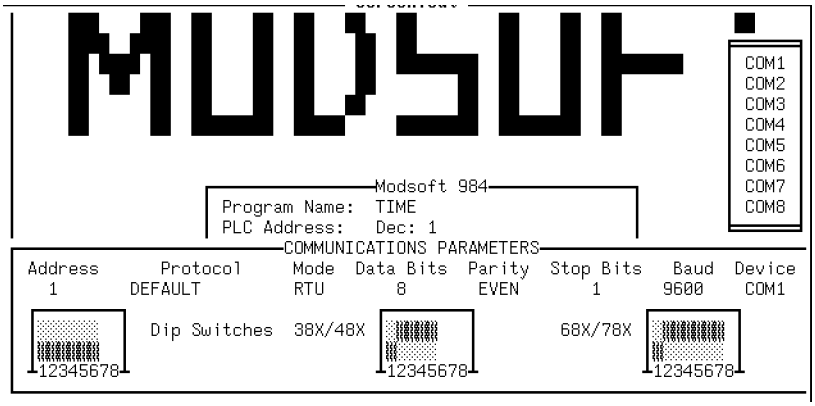


Figure 75 Communications Parameters Default Selected

3.12 (Offline) Program File Settings

The File Editor performs all the operations connected with loading and saving program, comment, configuration, reference data and reference symbol files. In addition, it can check the consistency between the actual loaded programs and the current configuration.

☞ **Note** Modsoft version 1.1 (and above i.e., upwards compatible) has a database structure that is different than the original release (1.07). You should archive your original program by creating a copy of it. When you upload (via Transfer) the original program and save it in a 1.1 environment, the program will no longer function in a panel with the original Modsoft release (1.07).

The difference between 2.0 and the 1.2 release is an automated environment file restore feature. Using programs developed under Modsoft 1.2 was explained in Chapter 1. The 2.0 to 2.1 change is expressed in the Read.me file in the modsoft directory.

When a program is developed using the Modsoft system, information relating to all primary aspects of the data is collected in these files:

- *Environment(.ENV)* This file contains variable user selections and file paths that allow the re-run of the same program with the same parameters already set.
- *LadderFile (.PRG)* This file contains the program that is executed by the PLC.
- *LadderCommentFile (.PCM)* This file contains descriptive text that has been entered to annotate the Networks and Segments (in DB IV format).
- *ReferenceSymbolFile (.RFS)* This file contains any symbolic references and comments that have been assigned to the absolute references (in DB IV format).
- *ConfigurationFile (.CFG)* This file.CFG contains the configuration for which the application has been developed together with the segment scheduler and the I/O Map.
- *ReferenceDataFile (.RFD)* This file contains the values that have been assigned to the references.

- *ASCII MessageFile (.ASC)* This file contains all the formatted ASCII messages.

Modsoft also produces a number of subsidiary files* that are dynamic in nature. These include:

Extension	Description
.CUT	Coil used table
.PCI	Network Comment Index
.RFI	Reference Table Index
.UPF	User Preference File
.PCT	Data base Tag file for program Comments

* See the first page of the Index , at the rear of this document, for a list of file extensions.

In addition, during the control logic conversion process the file “name.Map” is created and the flat ladder logic file with checksum information is created with a .IMG extension. The “name.Map” file is not active in programming operations as the above are, but it can be used to inform you about the actual number of networks created when the panel designed network is converted to controller logic, and give you the segment they reside in.

☞ **Note:** Write Protection on your project files causes problems in downloading if the protection includes .map or .cut files. Modsoft must write to these files to process the download. If this requirement is violated you will see a file opening message followed by a DOS environment error.

To view the Time.Map file, you could exit Modsoft to DOS and use the command *TYPE* (in the Program directory). The file format appears in the form:

Segment	Object	Network	Actual
1	P000	1	1
2	P000	1	3
2	P000	1	4
3	S001	1	9

In this example, the first entry in segment 2 under initial step P000 was 1 network before conversion but on the PLC level there were actually 2 networks (2 and 3) to execute. Because all programs or edit functions are performed on temporary files, a typical edit session would be to: Load a file set Change it Save it again

By working on temporary files, you can go back to a previous version of your program, you can select how to save, or call selected files that comprise a total PLC program.

When transferring State RAM from a controller, an additional file is created with a .REF extension and it holds the PLCRAM contents. This file has the same base name as the .RFD file.

A filename.CUT is made to provide the “Coil Used Table” data base.

The File Menu provides the functions in the next illustration.

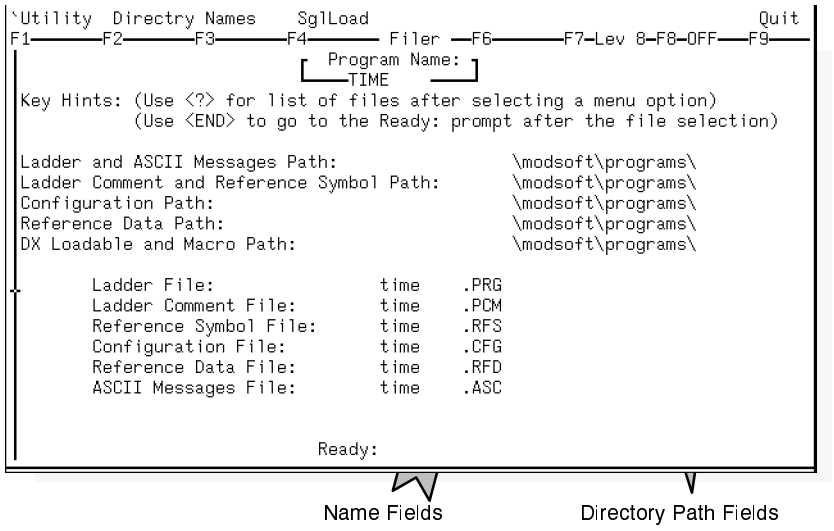


Figure 76 File Menu Directory and Current File Status display

3.12.1 Directory

Modsoft provides for maximum flexibility in the storage of PLC related program files. It is possible to have separate paths for the five classes of files:

Ladder and ASCII messages

(Offline) Program File Settings

Comments and Symbols
Configuration
Reference data
Macros and DX Loadables.

The "Directory" path setting function is used to set these paths. The default path is entered by the system at startup, from the PLC Program name.ENV file. You may want to set initial defaults for all files to be the same using this function to set:

\MODSOFT\Factory\ For example.

Whatever paths you select are retained in the PLC Name.Env file. It is highly recommended that only experienced Modsoft users change the directory paths from the default.

☞ **Note** When using the File directory functions, if you do not have a diskette in the drive set up in the DIR command line (i.e., a:MODSOFT\xxxx), the screen displays:

Drive not ready R(Retry), A(Abort) or C(Cancel)

If **A** is chosen, Modsoft exits to DOS
If **C** is chosen, the command is cancelled and you are returned to the screen
If **R** is chosen, the Modsoft system tries again after you place the correct disk in the drive.

Should you inadvertently press A, and lose changes, just restart Modsoft with the /s switch (modsoft /s) and most changes will be preserved in panel memory. You should immediately SAVE the changes.

☞ **Note** The paths selected need to pre-exist. This feature does not create new paths in your computer disk system.

After setting the paths, or pressing the <**End**> key, the cursor drops to the "Ready" prompt at the bottom of the screen. Press <N> if you change your mind or press ↵. If any of the entries in the path specification are not found, a system message is displayed.

3.12.2 Names

When you select this menu item a cursor is positioned in the first name field and you can edit existing names as required.

3.12.3 Sgl Load

This menu allows you to load single files.

The file names are entered without extension. The file type extensions are shown at the end of each input field. After selecting the program file type from the pulldown list, the file name and "." extension are displayed waiting your confirmation via the ↵ key.

You may continue the selection process by moving the cursor to the next desired file.

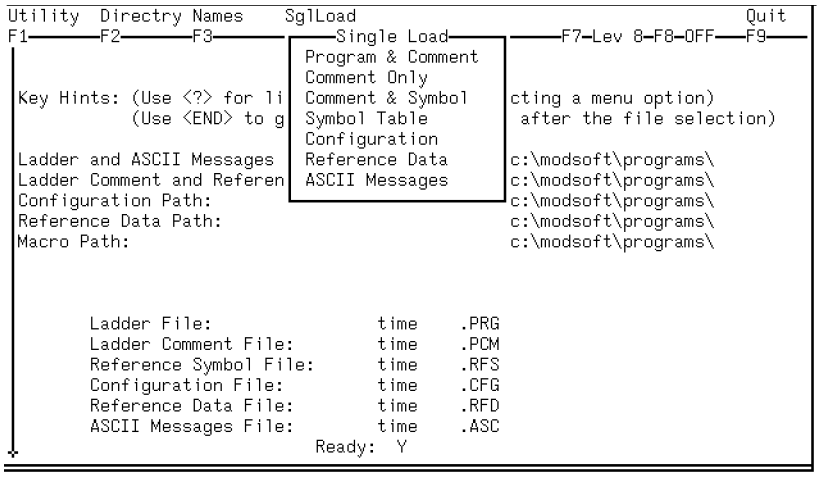


Figure 77 Single Load Pulldown Menu

When loading a program, the revision number from the program header is taken.

3.13 (Offline) Location Of Program Names

This feature informs the applications program what drive and directory to check for the .ENV files. The base default is \modsoft\programs\ which you can change using the .UPF (User Preference File) Parameter names_directory and files_directory to create a different path (See Appendix D).

The default path is entered by the system at startup, from the PLC default.ENV file. You may want to set initial defaults for all files to be the same using this function to set:

For example. \MODSOFT\PROGRAMS\ Paintshp

Whatever paths you select are retained in the PLC .Env file.

☞ **Note** The path selected does not need to pre-exist. This feature will create new directories on your computer disk system.

This is the feature that allows loading a program file set from removable media. I.E., you just specify A: for the drive. If the drive is not specified the device that the application is executed from is assumed.

☞ **Note** It is important to have a separate programs directory (folder) for 32 bit PLC's (e.g. the 424). This separation will help you segregate the User loadables between 16 and 32 bit types.

3.14 (Online) Select Program

The Online Select Program selection allows you to select the name of the program file-set you want to work with. When you enter the File name, the Reference Symbol and Network Comment file are loaded from the hard disk and Configuration files are transferred from the PLC to the Panel memory and the Communications parameters are set to the PLC Port. This activity is visible in a window on the display.

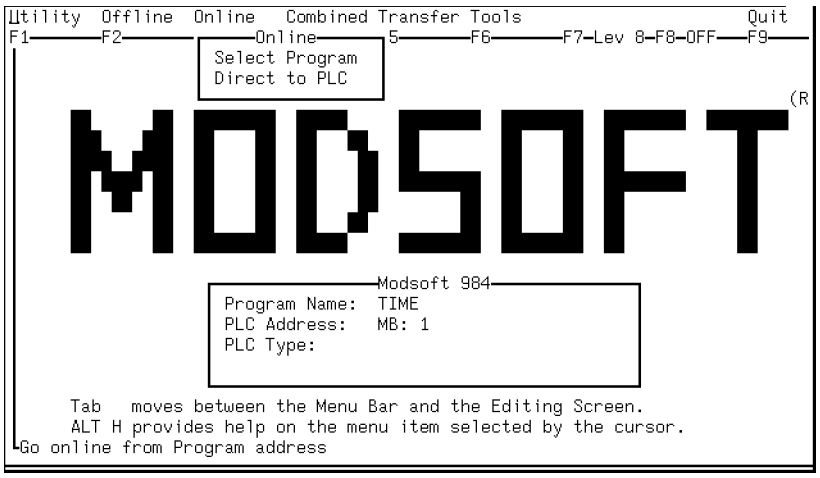


Figure 78 MODSOFT Online Main Menu Selections

3.14.1 Select Program

The Select Program function presents a list of program sets for your selection. The List is generated from the .env files in the directory path set using Location Of Program Names. The default is c:\modsoft\programs.

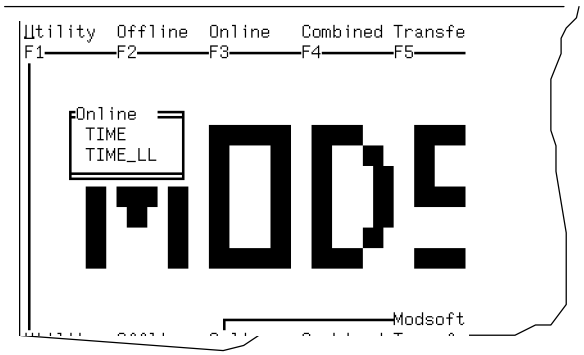


Figure 79 Select Program List display

With your program selected, the application transfers the Reference Symbol and Network Comment file from the Program path to the panel. This transfer reads the configuration into the panel so Modsoft has the proper PLC parameters. The transfer PLC to Panel progress is displayed as illustrated:

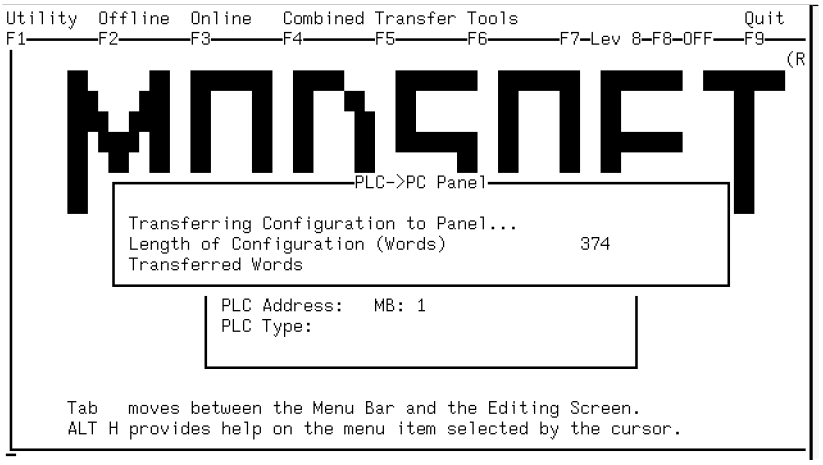


Figure 80 Online Select Program Configuration transfer

(Online) Select Program

If the PLC has been configured i.e., is capable of going into RUN, the files .PCM (Network Comments) and .RFS (Symbol) are loaded and the Segment Status display related to the selected network logic is displayed. The cursor is located in the first segment field and your first ladder diagram will be displayed with powerflow (see Network #1) when you press the return key.

> ☞ **Note** If transferring program to PLC you may start this program by answering “Y” to the Start Controller message

You are advised with a proper error message if the PLC is not configured.

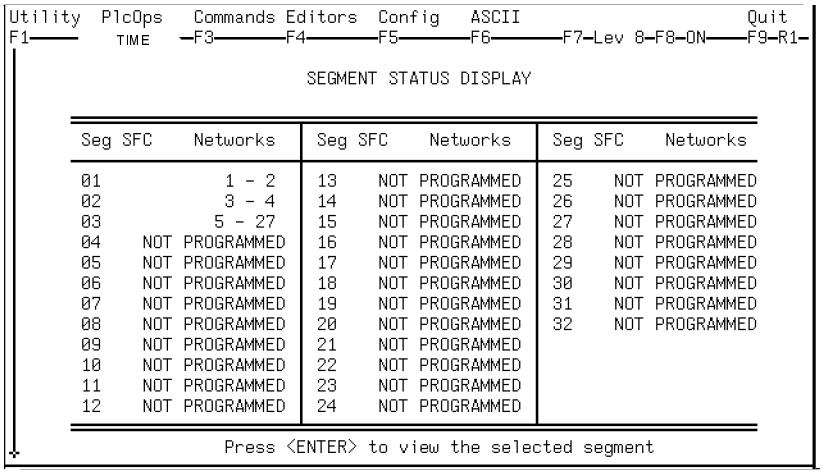


Figure 81 Segment Status Display

(Online) Select Program

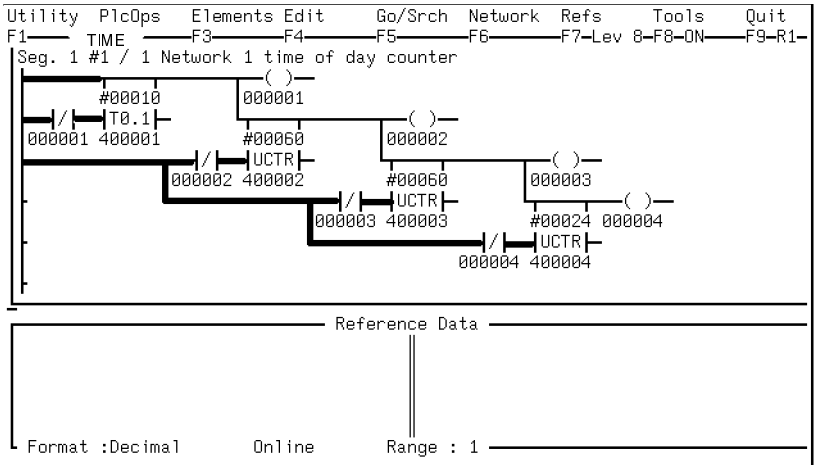


Figure 82 Select Program "Time" Result

You are now ready to create or alter your PLC contents (i.e., program).

☞ **Note** The field after the F9 key usually reports the status of the controller, followed by the slave address, as in R1 (running controller address1) above. In TCP/IP communications there is no slave so when using TCP/IP this field will read `Stcp` for Stopped or `Rtcp` for Running.

☞ **Note** Although this mode allows you to select programs: Symbols and Network Comments from the program name list, its primary purpose is to edit the program in the PLC while maintaining Network Comments and Symbols.

3.15 (Online) Direct To PLC

When selected, this function automatically places Modsoft in the Communication Parameter Display and when you exit from that display the PLC Configuration file is uploaded to the Panel internal RAM. If you executed this command immediately after a Select Program (Online) with the "Time" Example, your display will appear with Power Flow displayed but with no comment displayed.

Changes made to configuration or ladder logic are written to the controller ONLY. The first network is displayed either with an existing network or with a ready to edit display . The menu hierarchy within Online mode is parallel to that in Offline mode operations; however, some components are not accessible, as they are not provided in Online mode. In particular, the Symbol Table and Network Comments are not available. Additionally, some menus are adapted to the special functionality of Online mode. The Offline menus and selections, screen layout, special keys, etc. are valid for Online mode.

Tip: Modsoft times out in RTU mode if it was run in serial communications mode (ASCII or RTU) in a previous DOS window without closing that window (even if the previous Modsoft window operation had exited to DOS). When the previous DOS window is closed, the current windowed Modsoft can properly communicate over the serial port.

Direct to PLC requires you to confirm the communications parameters or change them as you need to, and presents the parameter display (the default parameters are always those of the last .ENV file selected) on the bottom of the display,

Note If you enter the Online Select Program or Direct to PLC mode and there is a program in panel memory, all configuration and program data are deleted from ***PANEL*** memory. A warning alerts you to this condition before action is taken and you are given a chance to save the files.

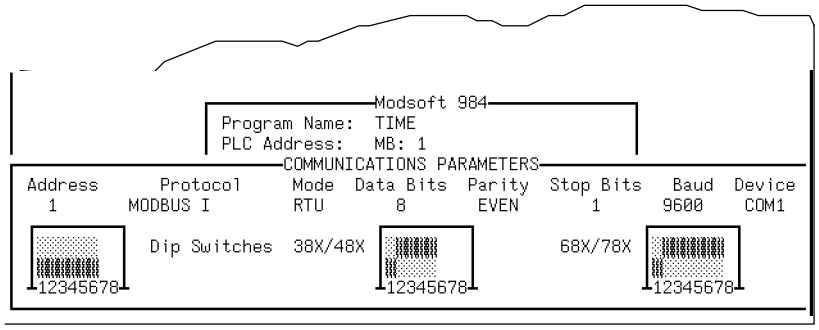


Figure 83 Direct to PLC select Communications Parameter display

☞ **Note** If you make changes to the controller program in Direct to PLC Mode and wish to "Save" the controller contents, you must exit Direct to PLC mode, select Transfer, issue a PLC to File, choose the Select Program Or New Program option and then do a save.

The Direct to PLC selection results in the application performing a transfer of the Configuration from the PLC to the Panel. The progress of the transfer is displayed as illustrated:

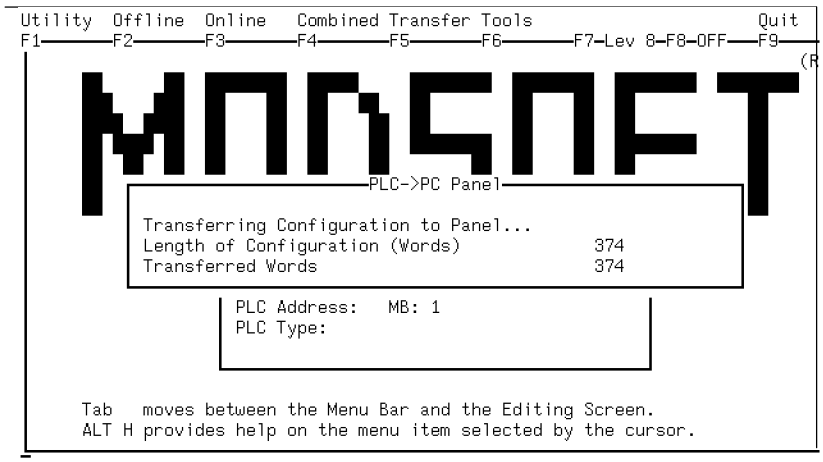


Figure 84 Online Direct Selected

This sequence results in initialization of the first ladder screen. You are now ready to create or alter your logic as defined in the Programming chapter.

Using the “Time” example illustrates the program functioning in a controller and power flow display on the panel. You can observe Power flow via **bold** blinking current paths after starting the controller. By placing the cursor over a logic element, you can see data change on the screen and logic Symbols you have assigned.

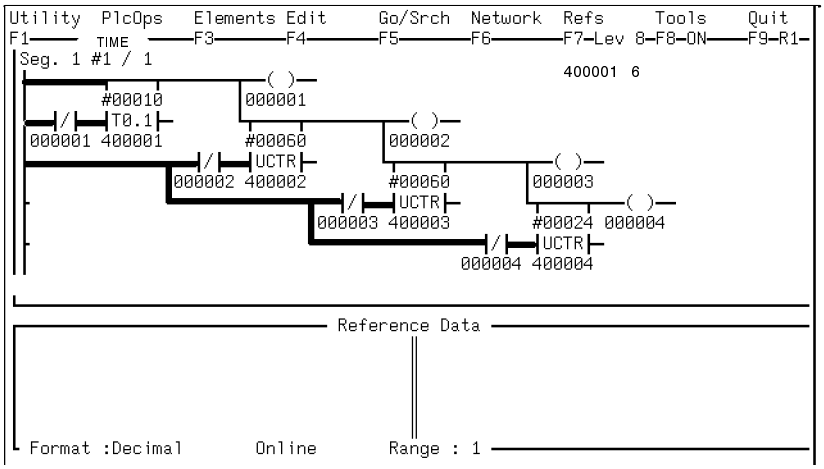


Figure 85 Display Power Flow Example

The identity of these components is checked when the Online mode is activated.

After configuration transfer, you are presented with the First Network screen. To continue the Online function you will use the edit and manipulation functions provided on the menu line.

From this entry point you can select, via the Ladder Menu function, from:

Utility PlcOps Elements Edit Go/Srch Network Refs Tools Quit

3.16 Combined

3.16.1 Select Program

When selected this function loads a named program file from its disk device into the panel internal RAM.

Combined mode provides the means for debugging a program during your project installation phase. The basic idea is that all changes made directly to the controller will be made to panel memory. Thus, it is possible to keep the program in the controller consistent with the one in the programming panel.

When entering this mode, a verify operation consisting of a logic checksum of the first 128 words of the configuration table is performed. If the panel and controller checksums are consistent, the program segment identification screen is displayed. If there is a verify fault, you are advised by a message on the display.

You are prompted to either

Transfer the program ***to*** the PLC or

Transfer the program ***from*** the PLC

After exiting Combined mode, you are prompted regarding the transfer of the State Table. If you choose **Y**es, the state is saved to the filename.RFD file. If you choose **N**o, no action is taken. Because the program could have been modified, you should finish the sequence by ***saving*** the program.

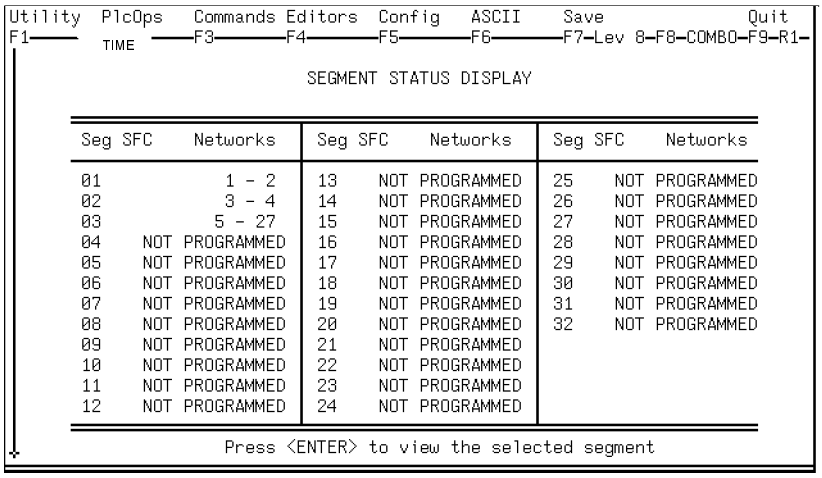


Figure 86 In Process Debug Mode Entry Screen

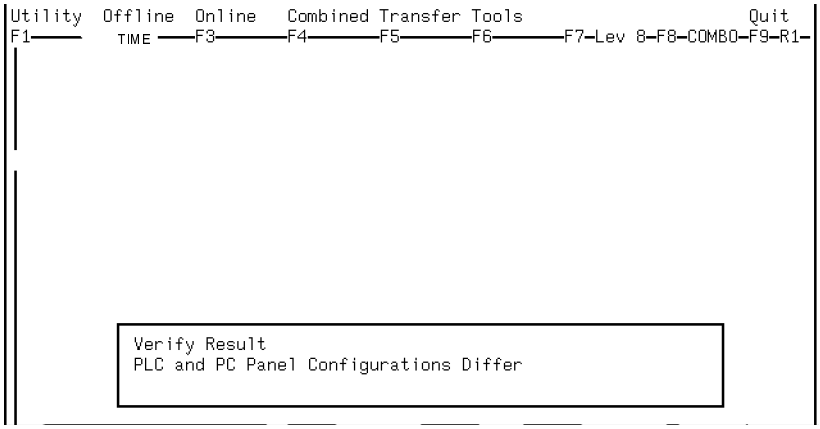


Figure 87 Debug Entry Mis-verify Screen

(Combined)Select Program

To demonstrate Combined Mode using the sample TIME program:

Step 1 Power up the system and initialize Modsoft

Step 2 Go to The Main menu and select "Combined"

Step 3 Select the TIME Program from the name list

Step 4 Select segment 1 from the Segment Status Display

Using the "Time" example, the logic display is shown in the Online figures illustrated in the previous pages. You can edit the ladder logic from this point.

3.17 Transfer

The functions available from the Main Menu Transferpulldown provide five selections:

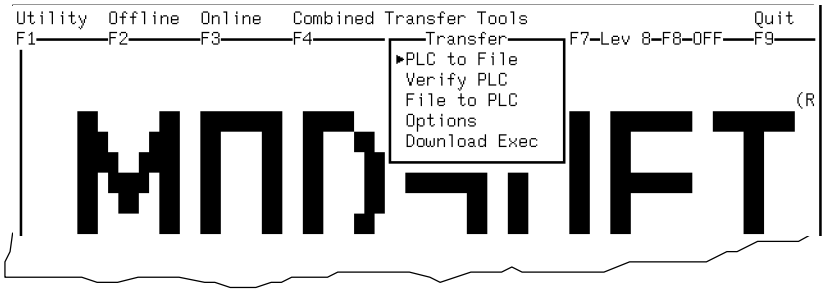


Figure 88 Main Menu Transfer Selections

As you begin to work with panel-generated programs and program edits, you must be aware that unless working “From File” or a save “to file” is done, all controller interactions are with the panel resident version. Therefore if changes test OK, you should save the program to a file, by using the Savefunction, or lose the changes offline.

3.18 (Transfer) PLC to File

3.18.1 PLC to File

The transfer of PLC software (including state RAM) to the panel is executed upon selection of either Select Program or New Program. Transfer to File saves the contents of a PLC directly to a file.

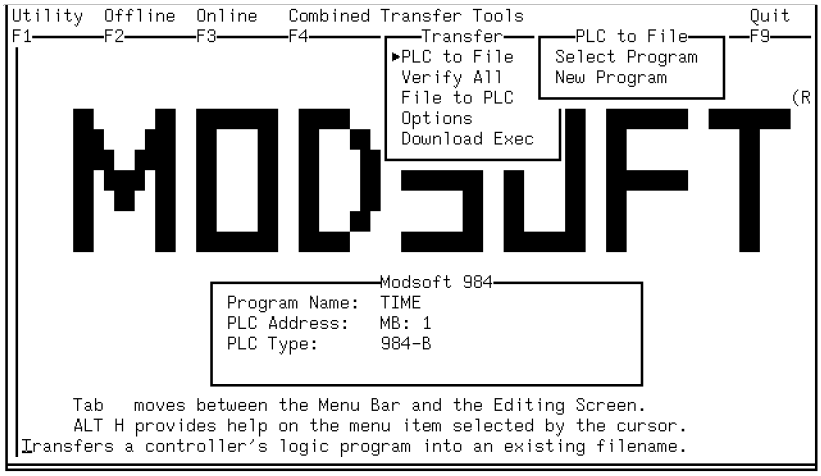


Figure 89 PLC to File

- When a project is transferred from the controller to a set of files, any .EXE loadables that are transferred are saved with the project files for subsequent downloading. These transferred loadables are in absolute addressing format, and are now applicable only to the project and to that PLC type. Any attempt to change PLC type on the program will potentially make the loadable un-usable. If this situation exists, you must delete the project's .msl file in your programs directory and re-configure the loadable.

Select Program uses previously set communication parameters. You are allowed to overwrite a previously created program with the contents of what is in the PLC.

New Program prompts you for the:

File Name
Directory Path
Communications Parameters

The contents of a PLC to a New file name is placed in panel memory. It is up to you to Save the file permanently to the hard disk.

3.19 (Transfer) Verify, PLC

3.19.1 Verify

This command performs a verify (byte compare) of the panel software (including state RAM) to the PLC software. As files are loaded, they are displayed on the screen. When completed successfully the cursor is returned to the pulldown menu.

If a miscompare is detected the display looks like:

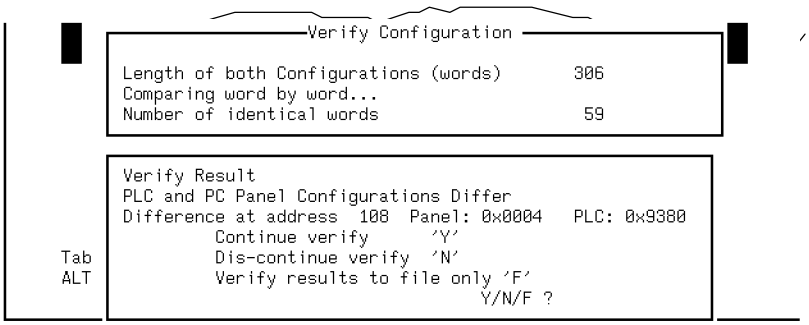


Figure 90 Mis-compareResult

The option “Verify Results to File Only” puts the miscompares in a file called Verify.Out but does not display them. This file is located under the Modsoft directory.

The N choice only puts the first instance of a mis-compare in the .OUT file

The Y choice displays all instances found.

At the conclusion of the comparison a FAIL display is posted.

3.20 (Transfer) File To PLC

3.20.1 File to PLC

The transfer of the program file to controller is executed upon selection. If the controller is running, you are prompted to STOP the PLC. Depending on the length of the programs involved the activity is indicated by the display. After the Transfer you are prompted to restart the Controller and the Controller Status Information is posted to the display.

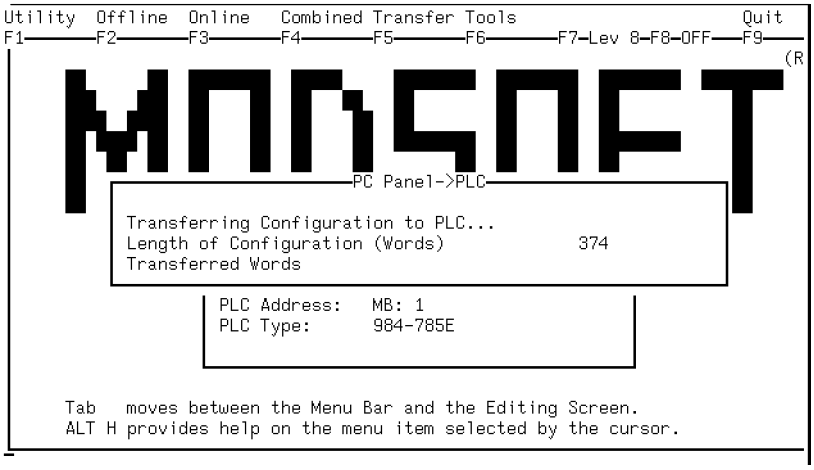


Figure 91 Data Transfer Process Indicator

☞ **Note** Whether you elect to `Quit` or back out of Modsoft using the **Esc** key You will be warned about saving changes even though *to you* there has been no change. The reason for this is the Modsoft conversion process, that prepares Ladder Logic code for CPU operation, is followed in all cases. This means that the software responsible for detecting change assumes that change has occurred and issues the user caution.

Warning! *Concept IEC PROGRAMS TRANSFERRED FROM PLC* which contain loadables, can only safely and reliably be transferred back to the ***same*** PLC.

You cannot edit such programs for Configuration Offline without the potential to create the problem noted above

You will be warned via the following interaction:

WARNING: Contains uploaded Concept (USL) Loadables **download ONLY to the same PLC!** If you are UNSURE, do a ConCept download, or reload the Original USL file OFFLINE

CONTINUE DOWNLOAD Are You Sure (Y or N)

3.21 (Transfer) Options

3.21.1 Options

You can execute the most used functions of the data transfers directly from the Options menu line that is displayed after you select the program set you want to work with.

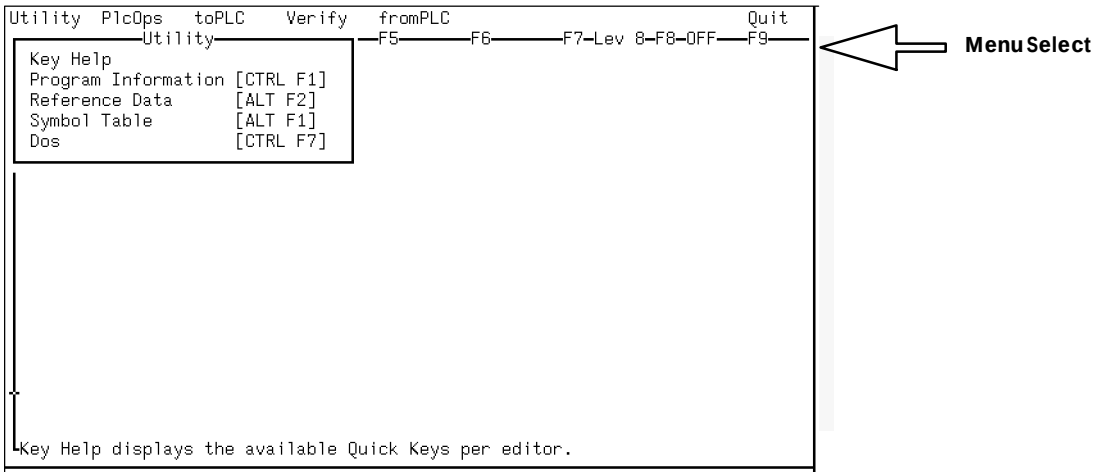


Figure 92 Transfer Options Menu Header

3.21.1.1 Utility

This selection provides the Key Help, Program Information and Dos entries defined at the Main menu level. There are however two additional entries:

Reference Data
Symbol Table

You can view Reference commentary statements and or view/edit Symbolic related fields. The purpose and scope of these two editors are documented in Chapter 7.

3.21.2 PlcOps

This entry was incorporated in version 2.2 to put the interactive functions:

Stop
Start
Single Sweep

in a convenient location to support the other Transfer functions.

3.21.3 To PLC

To PLC provides you with the ability to transfer panel files to the PLC in a number of different ways. You select the Transfer of choice from the pulldown illustrated below.

☞ **Note** If you start a transfer, then want to terminate the process, you can use the <**Ctrl + K** > keys, However since the processor may be servicing another task the effect of pressing this key may not be immediate. Keep trying.

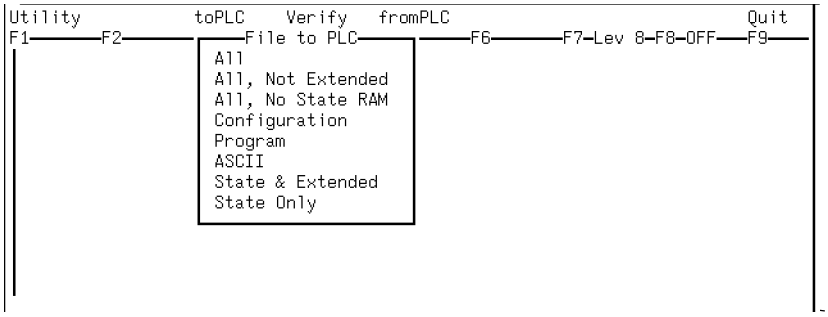


Figure 93 Transfer to PLC pulldown

- “All” Transfers the entire contents of the program file including State and Extended memory.
- “All, Not Extended” Transfers all as above but does not include Extended register references (6X)
- “All, No State” Transfers all as above but does not include the State Extended RAM or register contents.
- “Configuration” :Your defined controller configuration is written to the controller. All program data in the controller memory are cleared.
- “Program”: First your logic area of the controller is cleared, then the program is written to the controller, network by network.

(Transfer) Options

- “ASCII”: All the ASCII messages located in the programming panel are Transferred to the PLC.
- “State and Extended” transfers the state and the extended memory state (Values of References 0x, 1x, 3x, 4x, 6x) of references into the controller. This function is meant to be used for controllers that may have extended memory references like the 984-B, 984L, 785L or some “E” series controllers.
- “State Only”: If the state (Values of References 0x, 1x, 3x, 4x) of a controller was saved via the Filer I/O command, either from an upload or offline edits in the RDE, then the state of the references can be separately downloaded (if the references have specific contents) to the controller.

3.21.4 Verify

The Program Verify feature compares the panel to the PLC and, as of Modsoft 2.2, contains the following functionality:

- Displays any Logic Checksum differences between the Panel checksum found in the .IMG file and the checksum calculated by the PLC.
- Compares as many networks as possible, even if the number of networks in the Panel and PLC differ.
- When the length of a network differs, the number of nodes in the Panel and PLC network is reported and verifying continues with the next network.
- Displays network number, column, number of nodes in the element and element of node that miscompared.
- Display also includes name of the element (for example, BLKM or the graphical representation of contacts and coils) and reference number contained in that node of the element.
- When a BOC (beginning of column node) miscompares, the network and column number of the BOC that differs is displayed and verifying continues with the next network.
- All miscompare are written/appended with a time/date stamp to the VERIFY.OUT file.

- The user has the option to continue, discontinue or have the miscompare results written to the VERIFY.OUT file only after every miscompare.

Situations that cause Program Verify to discontinue verifying are:

- The Panel or PLC does not contain networks.
- Error in converting to a flat structure, such as a duplicate coil.
- Error in reading a network from the .IMG file or PLC.
- You select “D”, discontinue, on the Verify Option screen.

The Verify menu is illustrated below.

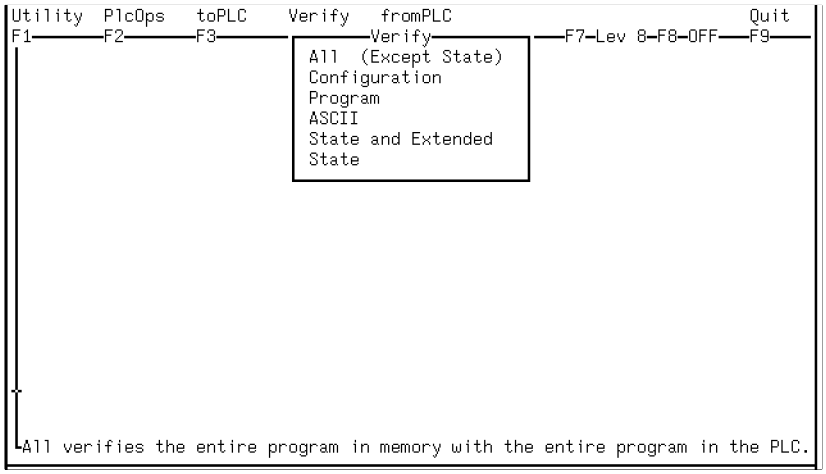


Figure 94 Transfer Verify Menu Options

- “All”: All data transferred to the PLC are verified, with display screens to inform you of the progress and results of making this selection.
- “Configuration”: The configuration data in the programming panel and the controller configuration data are compared. First the length of each is compared. Then, if identical, they are compared byte by byte. The compare sequence is stopped on the first miscompare. The result is displayed on the screen as illustrated.

The following screens show the effects of a miscompare and the info and options available to the user.

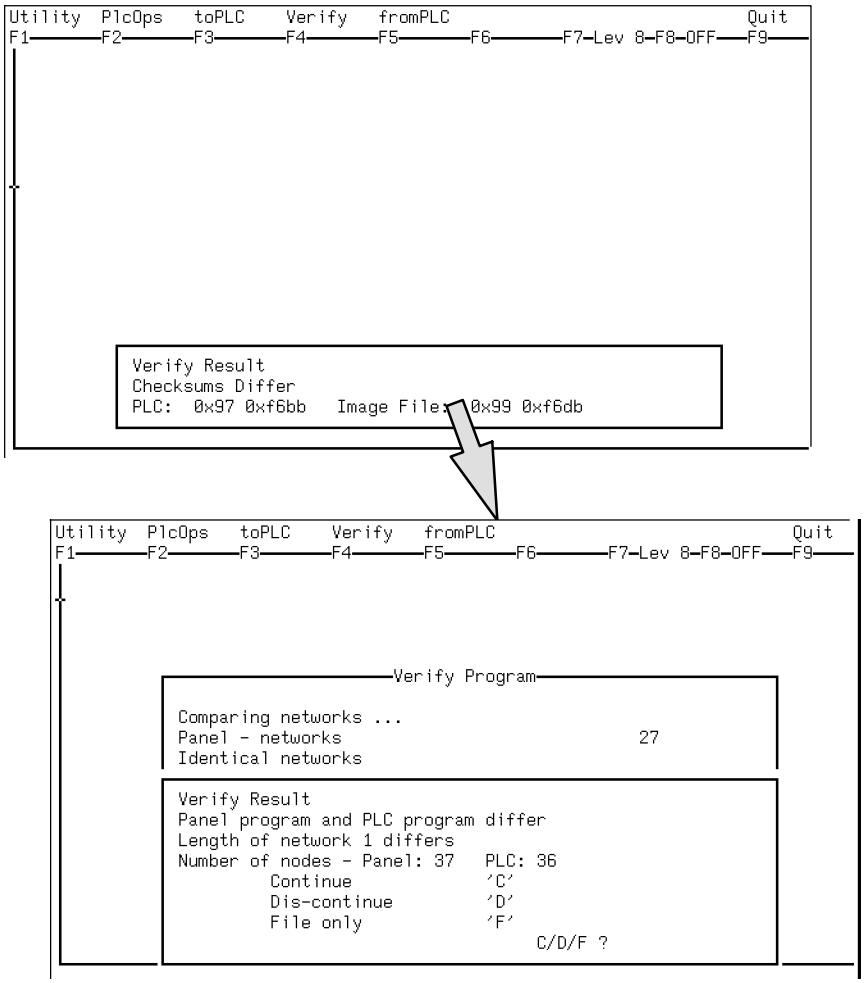


Figure 95 Configuration Verify Errors Using Time Example

- “Program”: The program in the programming panel and the controller program data are compared. First the number of networks of each is compared. Then, if identical, a network by network comparison is done. The result is displayed.

(Transfer) Options Verify

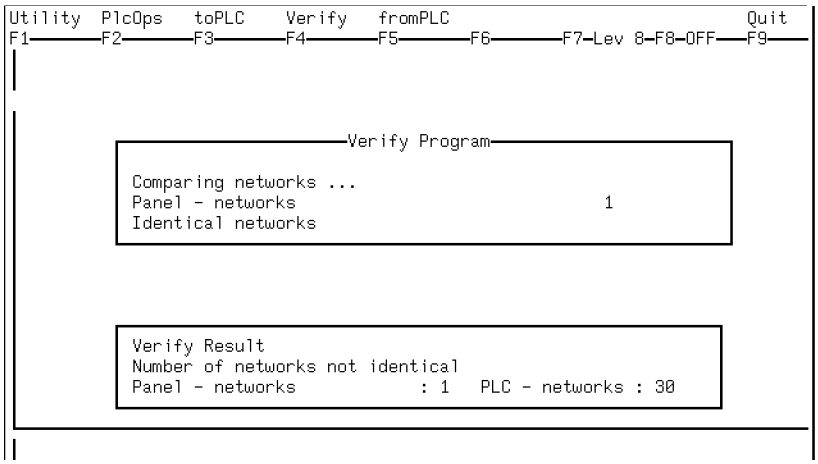


Figure 96 Verify Program Display

- “ASCII” Data in the programming panel and the controller ASCII data are compared. All the ASCII messages located in the programming panel are searched for in the controller ASCII message area and a size and byte information comparison of both is done.The result is displayed on the screen. If there are other messages in the controller, they are not compared.

- “State and Extended” This state verification includes the extended memory if configured.

- “State” This selection allows you to compare the Modsoft files with the state data in the Controller and displays a message stating if the state verifies or not. During the process a progress report is displayed on the screen.

3.21.5 From PLC

During this function you are prompted with a ***proceedY/N***?. This allows you to save any last minute changes you may have made in the panel memory.

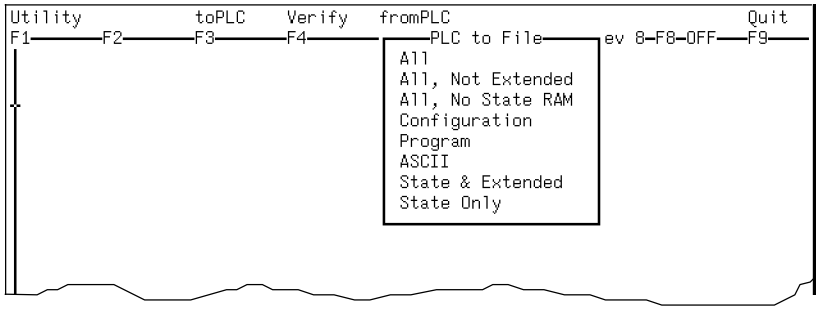


Figure 97 Transfer From PLC Menu Selections

☞ **Note** If SFC is used, it is converted to PLC format on the transfer. When you transfer data from the controller to the panel and that same program is there, there will be a “Comments Don’t match” error because of the original conversion. You continue from this error by simply pressing the space bar but the Network comment files are cleared.

- “All” Transfers the entire contents of the program file and PLC Image to the Panel memory.
- “All, Not Extended” Transfers all as above but does not include Extended register references (6X)
- “All, No State” Transfers all as above but does not include the reference state table.
- “Configuration” The controller configuration is written into the programming panel. All program data currently in the programming panel are overwritten; however you are warned and must verify the operation before the transfer is made.

(Transfer) Options

- “Program” The Program located in your logic area of the controller is written into the programming panel, network by network.

Caution If the configuration has DX’s, there is no check that the program DX’s are the same as the configuration. They could be different and alter the panel program.

If you have *two versions* of the same DX loadable, one in the controller and the other in the Modsoft DX Loadable Library, the PLC version will not overwrite the different version in the DX Library during the Upload.

- “ASCII” The ASCII messages located in the ASCII area of the controller are written into the programming panel.
- “State and Extended” Uploads the state and the extended memory state of references into the controller. This function is meant to be used for controllers that may have extended memory references like the Quantum, 984-B, 785L or some “E” series controllers.
- “State Only” When the controller is uploaded, you may choose to upload the state (content RAM value) of references separately.

When performing the “fromPLC transfer” and the DX loadable in the PLC does not match the DX loadable in the .DAT or .EXE file in the Macro directory, you are asked if you want to overwrite the file in the Macro directory. If you answer Yes, the file is overwritten and the DX loadable is configured. If you choose No, the DX loadable is not configured and you are prompted to save the DX in the PLC configuration to a file.

3.22 (Transfer) Download Exec

This entry in the Transfer function called “Download Exec” is a Modsoft tool that provides a download capability for those controllers or I/O modules that have Flash PROMS which can be upgraded in the field.

3.22.1 Using Download Exec.

When you make this selection, a pulldown is displayed for your selection of destination devices:

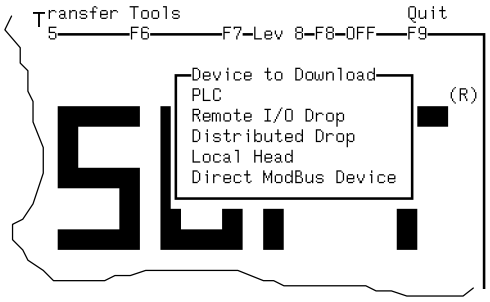


Figure 98 Executive Download Device Selection

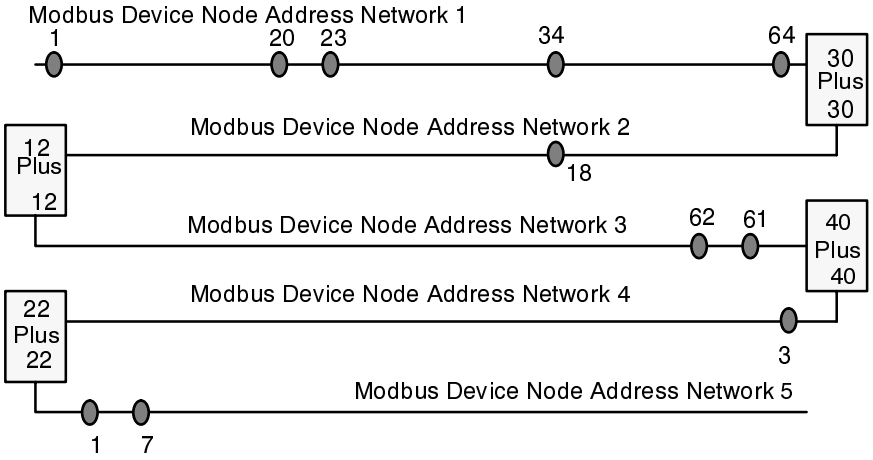
The prerequisites to performing this function are that you have the .bin format file that you want to download and obviously that the target PLC or Device be available.

3.22.1.1 PLC Executive Download Procedure

The first sub menu selection under “`Download Exec`” is:

```
PLC
```

The PLC choice signals that your intention is to Download to a controller and Modsoft responds with the display of the communications login parameters as previously illustrated. Although somewhat discussed earlier, the actual routing path was not discussed.You may choose to download an executive to a PLC which is a node on the Modbus Plus network. Sample routing parameters are illustrated by the following example:



In this example, there are 5 physical networks with 4 interconnecting Modbus Plus network hardware devices (the maximum allowed). Each network can have up to 64 Modbus addresses. The numbers inside the hardware devices illustrate that there is a distinct address associated with each device. This example is somewhat similar to what you see if you enter a ? in the Modbus Plus Address field of the communications setup display.

If you were doing a download to the Modbus device 7 on network 5 you would enter the following values on the communications parameters screen:

Address	Routing Address
7	30.12.40.22

☞ **Note** When only one network is wired, the routing address fields are used at the default value i.e. 00.00.00.00

☞ **Note** Download is not specified to work over a Bridge Multiplexer due to communications protocol differences at the interface.

If you want to review the device nodes on a Modbus Plus network (you have selected this communications mode) press the ? key while in the Address field of the communications parameters screen. The popup window lists the current configured network and you can make your selection directly from this list.

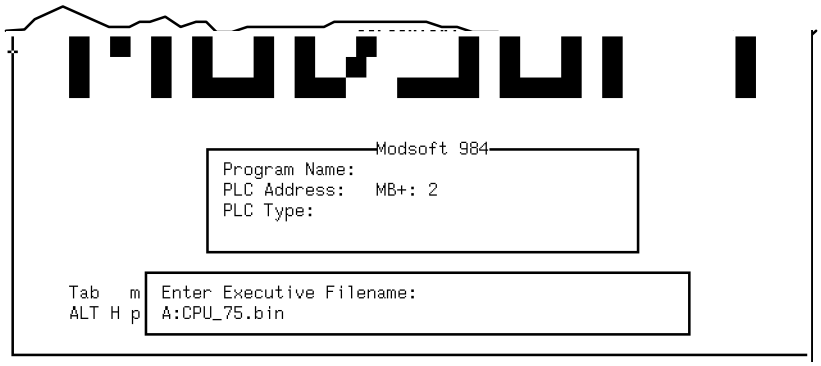
At the conclusion of your Communications parameter selection, the following single line is displayed.

```
Program contents will be lost after loading. Continue ? Y/N
```

If you do not have a current copy of the PLC program you should upload and use the `offline` file system to **Save** the program then enter **"Y"** to proceed.

To proceed enter "**Y**" at which time you see `"Enter Executive Filename"`:

The illustration below is an example that indicates the login port is Modbus Plus, it has an address of 2, and the path portion of the filename directs that the file is found on device A:



If the controller selected for the download is currently in the **Run** Mode, You are directed that the:

```
Controller must be stopped - STOP CONTROLLER (Y/N)
```

To continue the download press **"Y"** and observe the Executive Download Information Header display. You are required to verify that the correct parameters are given. This pause allows you to discontinue the operation if the current *exec*.is already up to date or if the exec. you are going to download is not up to date.

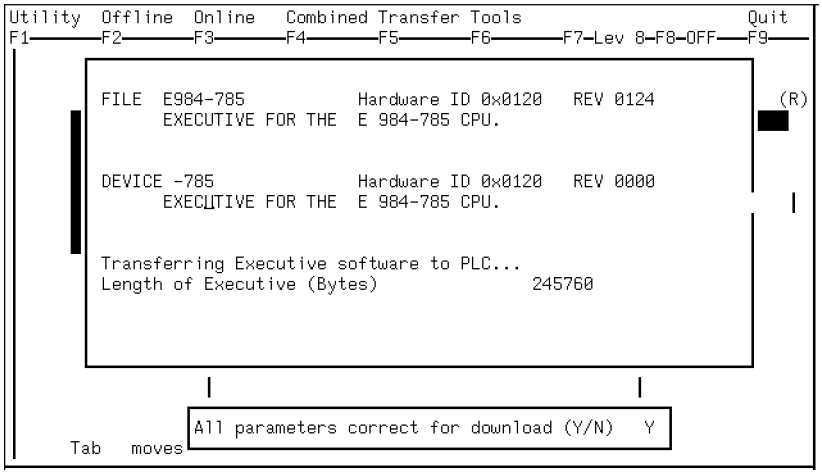


Figure 99 Example of Download Information Header

If the target controller is bridged to a network, the link is disabled; otherwise the Exec area in the PLC is Cleared, then actual data transfer proceeds (these 2 events are posted to the last display line in the header information screen while the event is in process). At this point and a normal condition is signaled by a series of 3 one second blinks on the RUN Light followed by a 3 second Off period. When the download is complete the link is re-established and the menu cursor returns to the Transfer pulldown. The controller can be restarted after downloading the PLC program and using the `PlcOps` **Start** command under the `Transfer` menu.

3.22.1.2 Download To Remote I/O drop

After setting the communications path you must define the **drop** and **slot** number of the head (140-CRP-93x-00). A display box with these two fields are displayed to prompt you. Remember drop 1 is the Local drop therefor this number is invalid for a remote I/O drop. The maximum is also limited to 32.

The maximum Slot value is 16.

After entering the drop and slot values the process continues (as in the PLC description) from the point of entering the Executive filename.

3.22.1.3 Download Distributed Drop

If you select this Transfer type, only the drop must be specified. The process continues (as in the PLC description) from the point of entering the Executive filename.

3.22.1.4 Download Local Head

If you select this Download type only the Slot must be specified and the data is transferred over the backplane. The process continues (as in the PLC description) from the point of entering the Executive filename. Modules that can be downloaded in this configuration are the 140-NOM-21x-00, a one or two channel head or a 140-CRP-93x-00 which is a 1 or 2 channel Remote I/O head, or Network Option Modules (NOE).

3.22.1.5 Direct Modbus Device

If you select this Download type, only the address on the network needs to be specified before continuing the process as above. In this configuration the download is over a point to point cable between the panel and Modbus device such as a Micro Hand Held Programmer.

3.23 Tools

These entries in the Main Menu Tools selection pulldown are defined in stand alone chapters:

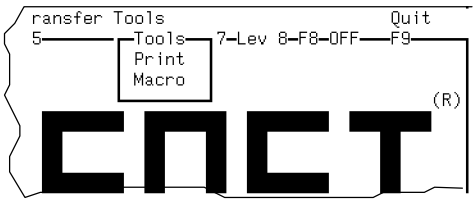


Figure 100 The Tools Selection Pulldown

Print details the options available to you for documenting the program you develop and Macro provides a technique to reduce similar program sequences by macro construction.

Chapter 4
Major Mode Support

- When you select Offline, Online, or Combined functions from the Main menu you are presented (within various menu structures) with file selection functions to start you in the programming process. With proper file selection, these major modes make available a large assortment of conditions you can use throughout the programming sequence including the manipulation and movement of configuration and programming data.

  These functions are presented here so the issue of Configuration, I/O Map and specific Logic Programming, in general, are not overly burdened with extraneous detail.

- Commands and Editor functions that are common among the operating modes are only defined once i.e., the Offline "network list editor" is not repeated for Online or Combined unless the results are different.

4.1 Commands

The Major Programming Mode selections from the main menu provide you with Command and Edit features at the Segment Status Display level.

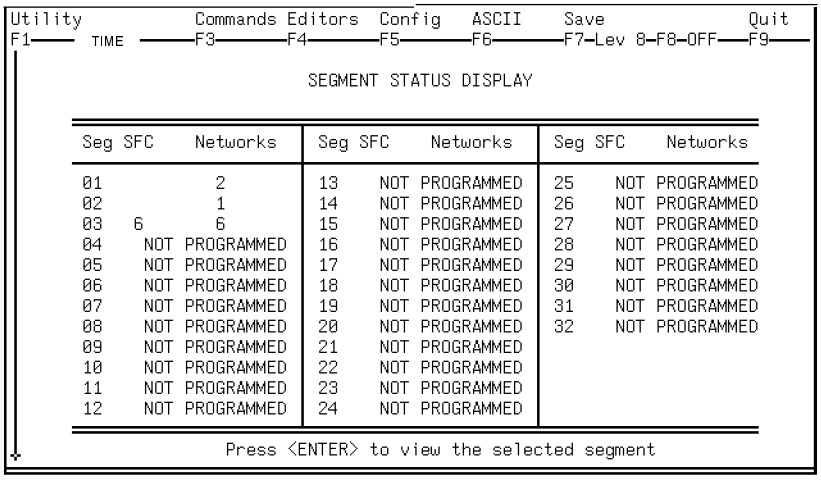


Figure 101 Offline Segment Status and Function Select Menu

The network you design can be annotated to give the network symbolic titles and element symbols. For ease of programming, one or more networks can be reused in several parts of the program or in different programs by creating libraries that can be understood as a collection of program fragments.

Parallel to the program and support editors, you can assign global symbols to references, using the reference symbol editor. This allows programming with symbolic operands. With the reference data editor you can define values for references, store them on file, read them from file, transfer them to the controller, and transfer them from the controller. Reference and Symbol descriptions are presented in a separate Chapter.

4.1.1 Programming Initial Status

The initial screen provides you with a summary status of the segment usage and identifies Sequential Function Chart (SFC) segments if there are any. The following is an example of the results obtained when the “TIME” example is resident in the panel.

When you first select a Program function, the first Screen you see is the Segment Status Display that applies to the program in the panel. The Status display also offers you a Command

menu selection for program manipulation. If you are creating a new program all segments are at the default condition and the program segment cursor is started at Segment one.

If the segment contains networks, you can place the cursor on that segment and display the network Logic by pressing the ↵ key.

☞ **Note** As a programming convenience some functionality is duplicated in various pulldowns. If the section you are using has pulldown menu entries without a text description, check the table of contents to determine the page where the information is found.

4.1.1.1 Commands

The Offline Program entry menu entry `Commands` pulldown offers you the ability to locate and manipulate networks both program and documentary in nature. The Commands functionality is more restricted in the other two programming modes where:

Online offers Search in Program
Combined offers Search in Program and List Used Macros.

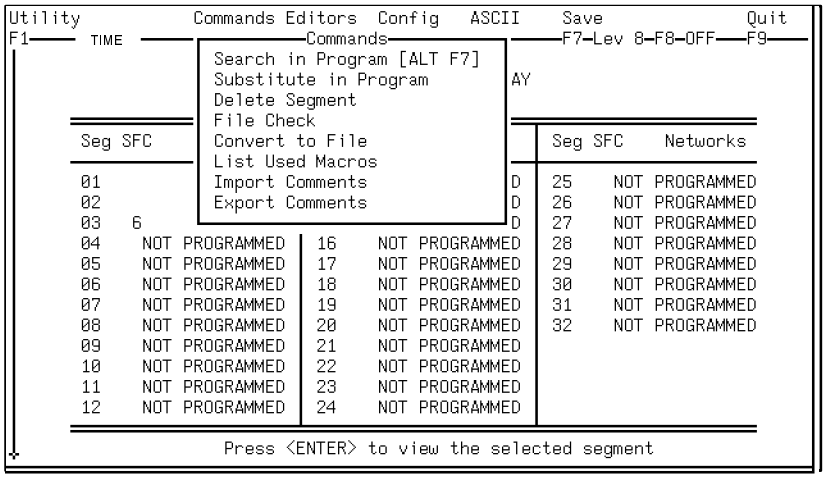


Figure 102 Offline Programming and Documentation Command Entry Point

4.1.2 Search In Program

This function searches through the program (**ALL SEGMENTS**) for a specified pattern and creates an ASCII file called **SEARCH.LOG** , in the Modsoft directory. These patterns can be references, symbols or network comments. The Search in Program does not search for func-

tion blocks, coils or relay graphics. The search is done segment by segment starting with Segment 1. The pattern is searched in the assigned segment title and segment comment. Then all associated networks with their titles and comments are also searched. Search data includes:

- 0x, 1x, 3x, 4x and 6x registers
- Reference symbols
- Network Comment Titles
- Segment Comment Titles
- Object Comment Titles

For example, searching for reference 00004 results in a display similar to the one shown in the figure below. To find and display the next occurrence press “Enter”. This data is related to the ladder networks in Segment 1.

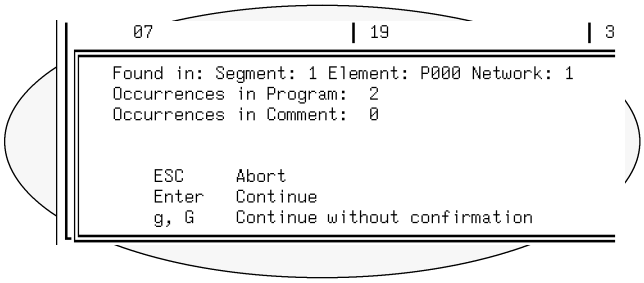


Figure 103 Program Search Result

The search pattern may contain only **letters, digits and underscores.** The system reports how many occurrences of the search pattern have been found and what the locations they were found in. This report can be answered by <**Enter**>, <**Esc**>, “**g**” or “**G**”. <**Enter**> causes the program to go on, <**Esc**> aborts the function and “**g**” or “**G**” tells the program to go on without any further request for answering. The occurrences that are found are also written to the text file SEARCH.LOG. You can print this file to provide a detailed report of your matches. The shorthand for this function is <**Alt F7**>

4.1.3 Substitute In Program

This function is quite similar to the Search but here you define a search string and a substitute string. If a match is found, this part of the string is substituted by the substitute string, (if it has been found in a segment title or comment). In the substitution string only **letters, digits, underscores and the # sign** are allowed.

The system reports how many substitute operations have been performed successfully and where the substitutions have taken place. This report must be answered as described in the search function.

> **Note** Substitute in Program does not substitute function blocks, coil, or relay graphics.

4.1.4 Delete Segment

Put the cursor on the segment you want to delete; then select this menu entry. You will see a warning prompt on the display as illustrated.

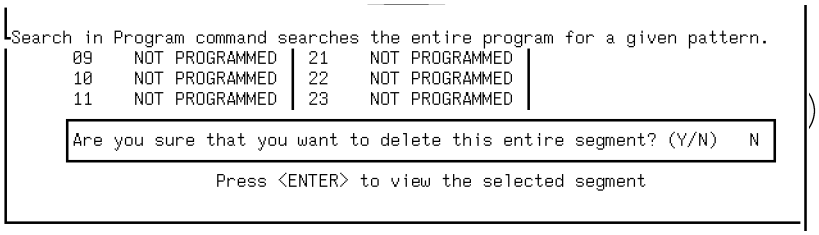


Figure 104 Segment Status menu Delete Segment

4.1.5 File Check

This function allows you to determine if a program matches the configuration it is based on. Syntax is checked for Range values, valid DX Duplicate Coils, Nodes etc, but does not include Macros or SFC.

Any errors found are shown as they appear. To acknowledge press the <**Return**> key or cancel with the <**Esc**> key.

An error summary is found in the file Modsoft.ERR in the Modsoft Directory.

4.1.6 Convert to File

This function converts a Program logic structure to a file format. The SFC element forms used for SFC programs are deleted and you are warned accordingly. You can keep the present program filename or rename the file. During the convert process, a summary of memory use is calculated and provided to you as a system message which gives the number of program words free (remaining).

4.1.7 List Used Macros

This function allows you to review existing Macro file names. The next display shows the one Macro that exists after creating or loading the “Time” example. The function shows all the MACROS that are being used by this particular program loaded in the Temp Files.

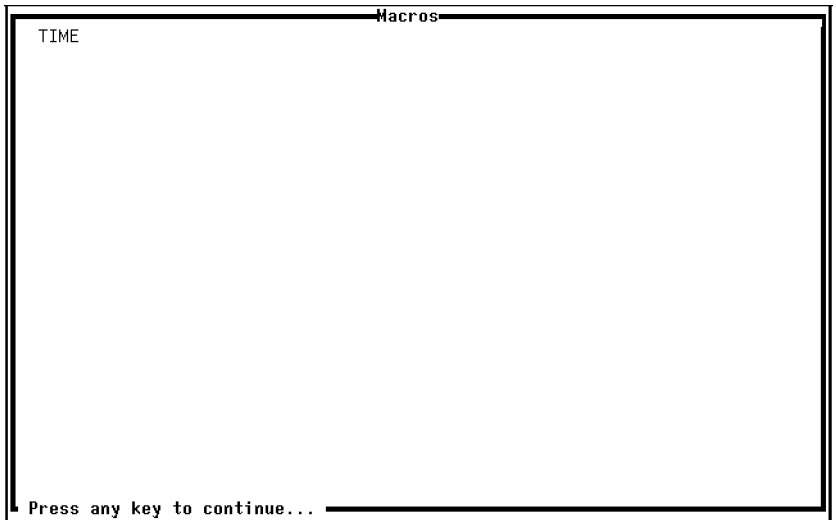


Figure 105 Available Macro Listing

4.1.8 Import/Export

The “Import” function, operates on standard text file, which must have the same format as an Export file. It can be used to rewrite the segment comments and network comments of the currently loaded program. For example, part of the Export file “Time.PCM to disk A:” looks like The next illustration. If you import data created with an external text editor it has to have the same format and delimiters as the Modsoft created text. Only as many networks as exist in the current program are imported.

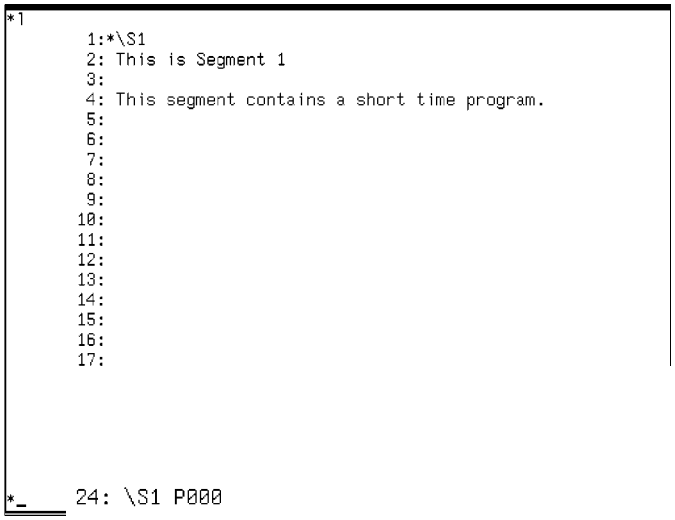


Figure 106 Export File Format Example

Using the "Export" function, you can export the segment comments and network comments of the currently loaded program to a standard ASCII text file. To export symbolic reference comments, use the symbol table export function. The export file contains the original program name and (for every entry) the actual segment number, and the network number with that segment.

4.2 Editors

4.2.1 Program Mode Function

Of the Seven Editors available in Offline, Ladder Diagramming and Sequential Function Chart (SFC) have independent Chapters devoted to describe them. The remaining Editors (4) are described here and apply equally to all programming modes except Program Library that is available only in Offline.

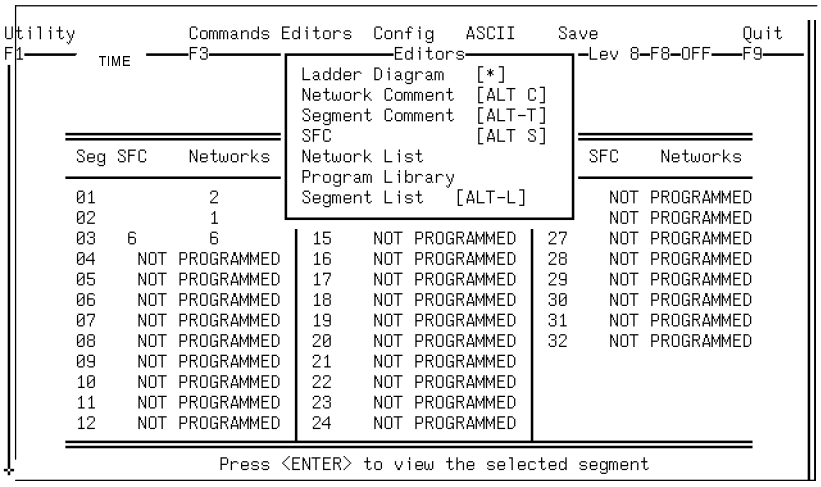


Figure 107 Offline Editor Select Pulldown

4.2.2 Network Comment Editor

This function is selected with cursor placement, function key sequence or quick key [Alt C].

This editor is a special text editor that allows you edit the title and comment of a network. When a network is inserted in the program, an empty title and comment are automatically assigned. The shorthand key is <**Alt C**>. The network comment menu contains the following items:

Utility Edit Goto Quit

If there is a network indicated by the Segment Status Display, you can Edit related comments. In this example you can see the comment and in the sub diagram you can see the Menu for Edit functions and the Goto Network selector prompt, all of which you will use when generating Network Comments. In this illustration the program contains SFC. When you have SFC OFF or are in Online Mode the GoTo prompt has only the Network and Title fields.

The Segment and SFC object comment editors have similar features.

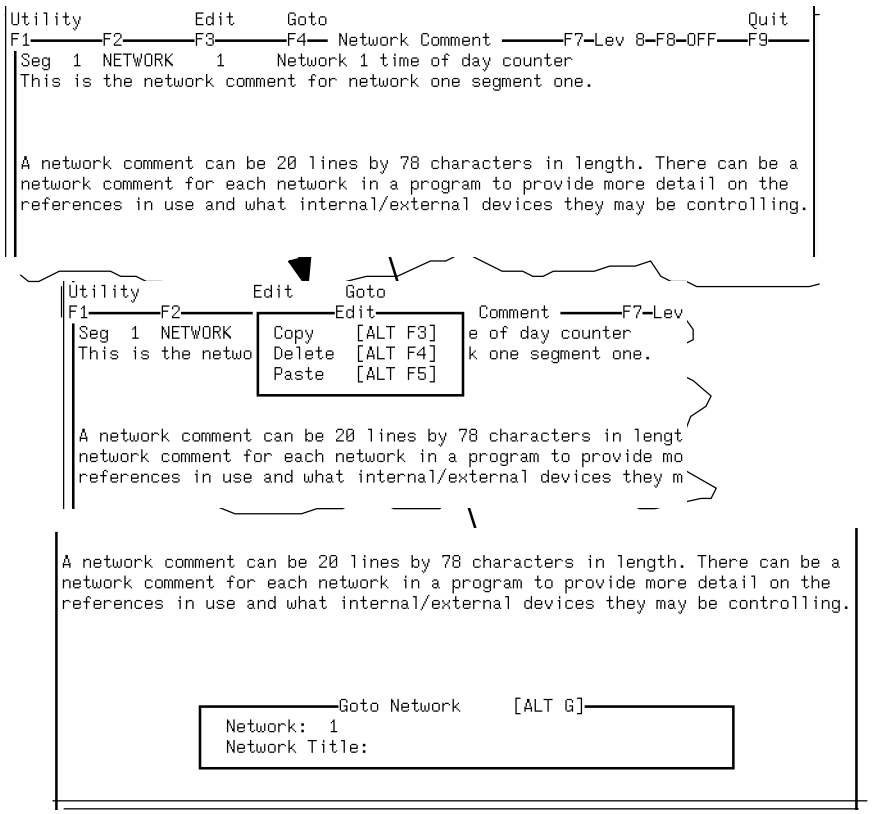


Figure 108 Example of Network Comment

4.2.3 Edit

This menu selection offers a `Cut and Paste` operation and a delete buffer function. When you select the Copy command you define the length of the comment by moving the cursor until you find an end point. The selected area is displayed in reverse video for easy identification. Terminate the Cut with a return and relocate the cursor to where you want to relocate the text then go to the menu and select Paste.

The `Delete` selection is defined with the cursor and reverse video in the same keys as above. When the area has been defined to your satisfaction just press the **Return** key.

4.2.3.1 Editor Key Use

The cursor is initially placed at the first character of the comment. From this location you may press:

Control →	Control plus Right arrow moves the cursor to the *start* of the next word.
Control ←	Control plus Left arrow moves the cursor to the *start* of the previous word.
Control End	Control plus End moves the cursor to the *last* character position on the screen.
Control Home	Control plus Home moves the cursor to the first position in the *Title* line.
Home	Puts the cursor on the *first* character of the current line.
End	Puts the cursor on the *last* character of the current line.
Insert	Shifts characters on the *currentline* until line wrap is required.

4.2.4 Segment Comment

The Segment Comment display is illustrated next. This display has a similar editor format as Network Comments defined above.

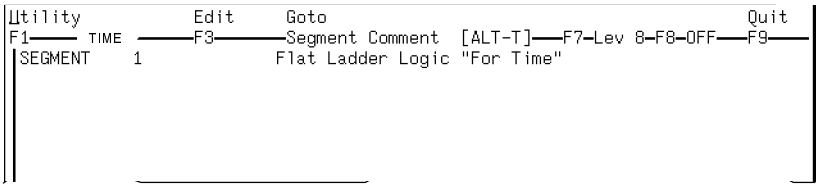


Figure 109 Segment CommentSelected

4.2.5 Network List Editor

This function, when selected from the Editors Menu, opens a window containing the associated networks for the step or segment the cursor is placed on. A new “network cursor” appears, which is moved up and down to select a network. In the Netlist mode, networks can be inserted by the command “`Insert`”. You can use the list as an index into the available components that you can select for Copy or Delete and Paste operations which increase productivity.

The netlist window opens with a width of 33 characters and the height of the screen.

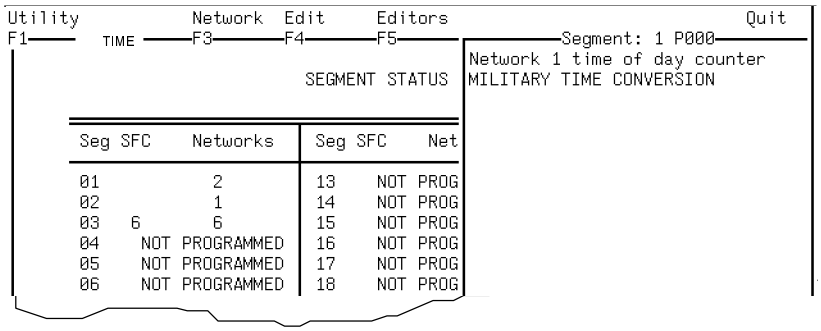


Figure 110 Example of NetList Window

If a network has an assigned title, this title is displayed within the Netlist window otherwise the network number is displayed. The netlist mode is left by “Quit” option or pressing the <F10> key.

To copy an entire network (or more than one) select the Copy command while in the Netlist window then put the cursor over the networks you wish to copy. Terminate the selection with a ↲. You now place the cursor at the location in the list that you want the copy to appear, then select the Paste function. You will see the list is updated and when you leave the window the count in the Segment Status display is updated to reflect the new number of networks.

4.2.5.1 Network

Goto Network

This command serves as a method to access networks that are not adjacent to the current network, but are within the current Segment. You can specify either the reference network number in the step or a search pattern, which is compared against the network titles. If the search pattern is not found, the search starts again with the first network. It stops, if the current network is reached again. The found network is displayed in the work window. The special keys <**PgUp**> and <**PgDn**> bring up the previous and next network of the current step into the work window of the ladder diagram editor, if there is any and <**Ctrl + PgUP** > or PgDn can be used to change to different segments. The shorthand key is
<**Alt G**>.

Insert Before Network

This command inserts a new network before the current one. The current network and all that follow are shifted to the end, re-numbered, and the inserted empty network is shown on the

screen. For the inserted network, an empty title and comment area are inserted. The shorthand key is: <**Alt I**>

4.2.5.2 Command

The "COPY" command copies a node, nodes, complete networks, multiple networks or, at most, a complete segment. A copy frame is created by using arrow keys or the cursor keys, < **PgUp** >, < **PgDn** >, < **Home** > or < **End** > keys. A carriage return will copy the highlighted area into a paste buffer. The shorthand command for Copy is < **Alt F3** >.

The "DELETE" command creates a delete frame by using the arrow keys or cursor keys < **PgUp** >, < **PgDn** >, < **Home** > or < **End** > keys, as above. The frame is displayed in the network window by a highlighted region. When the Enter key is pressed, all networks within the highlighted area are deleted. The deleted networks are placed in a temporary paste buffer in case you wish to recover them. The shorthand command for Delete is < **Alt F4** >.

The "PASTE" function is allowed only if the paste buffer contains data from a prior Copy or Delete command. The contents of the buffer are inserted in the network list at the actual cursor position, and all other networks are shifted to the end with re-numbering. The shorthand command for paste is < **Alt F5** >.

☞ **Note** The Copy, Delete and Paste functions ALL use the SAME buffer.

The "OFFSET" command lets you transform some operands within the network list to other addresses. You are prompted for a range of references and an offset value of the modification. If the offset addresses are valid, mark the number of networks to be changed by using the same keys mentioned in the copy and delete commands. The number of replacements is reported when the command is executed. The shorthand command for Offset is < **Alt F6** >.

4.2.6 Program Library

Any previously created program can be loaded as a library. At entry of the library, its Segment Table is visible. You can use the library feature to merge multiple programs into one program, copy specific networks or complete segments. Changes to the library program are not saved. Paste buffers can be loaded with part of the library program. These paste buffers can be inserted into the present program. If you leave the library program, control returns to the present program. The initial entry screen for this function is illustrated next.

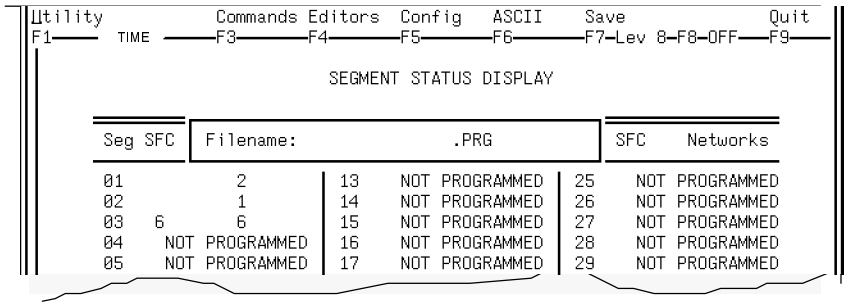


Figure 111 Zoom to Library Entry screen

If you copy a Network list from a Library file and paste it into the original program in memory, the Network comments from the Library program file will be available; however, the symbols associated with the references will not be.

If you copy several nodes of a network in a library program file then paste them into the original file (in memory), the symbols for the nodes will not be available (unless the references are already in the symbol table).

Warning! *Programs imported from another Programming Panel* may have been uploaded from a PLC and may thus contain exe loadables. In the event that the destination Panel does not already have a Library copy of the loadable, Modsoft will create one from the imported program. If this program came from a PLC, the created library copy of the loadable will already be absolutely relocated, and can only be used in PLC's matching the memory type and allocation of the execution buffer of the "Donor" plc. But the allocation of the execution buffer is not usually known by the user, thus making this program copy susceptible to improper operation.

ISSUE:
Transferring a program containing uploaded exe loadables to another PLC which has a different memory organization will most likely prevent the PLC from operating, but more subtle effects are not impossible, such as operating for a moment, but performing unpredictable actions.

REMEDY:
The only safe way to get a program from another panel PC is to insure before copying it, that the loadables in the saved program are not already relocated. This can only be done by editing the program's configuration offline in Modsoft, and then saving it; for example, delete and re-enter the bottom loadable on the list. This will recreate the program's usl file from the library copies. If there is any doubt of the library copies having been themselves substituted from uploaded programs, re-install the loadables from the distribution copies.

4.2.7 Segment List

The Segment List serves two purposes:

First it enhances the segment status display by providing a brief description of the Segment. You can display networks by direct selection from the List. This allows you to quickly review the ladder logic, from which you return to the Segment Status Display.

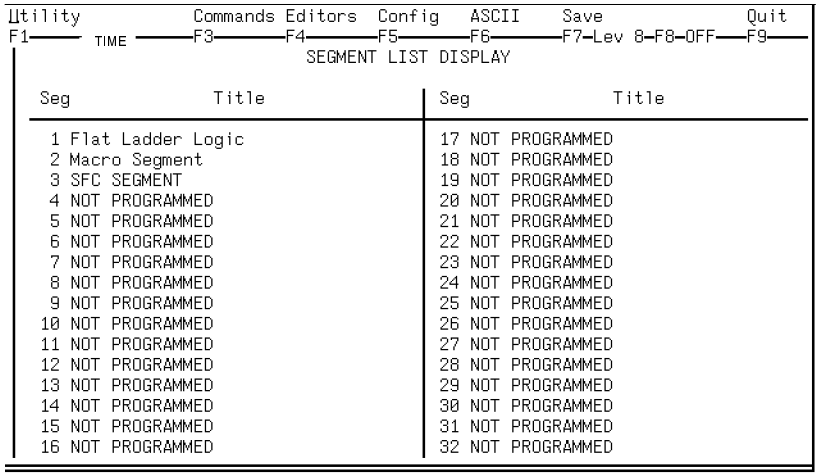


Figure 112 Segment List Display Using TIME Example

The second feature of this editor is that you are allowed to customize the segment title to your needs. For example, You can select the Segment Comment Editor and change or modify the comment field. In the example “For Time” is added to each of the 3 Segment titles. This edit session is related to the segment you place the **Segment List Cursor** on i.e. that becomes the segment to be edited on the Segment Comment editor.

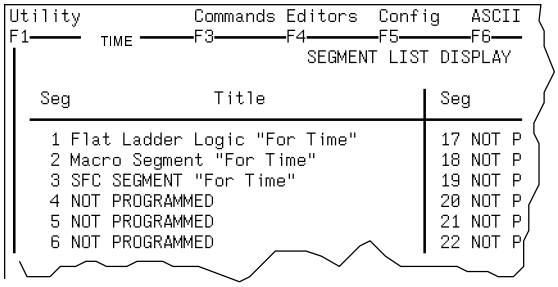


Figure 113 Example of editing the titles

Chapter5
Modsoft PLC Configuration

- Controller Configuration in Detail
- Product Support via I/O Map
- Segment Scheduler
- Loadables
- Configuration Extension

5.1 Configuration

System configuration has some far reaching implications because it affects the overall operation of the controller. The configuration must specify all the information that is controller specific, as well as the general information required to allocate memory, designate the quantity of various data types, and configure the input and output I/O sections. Before running the PLC, the first time, you must edit some basic items in the controller section such as PLC type and memory. This section of the configuration screen is available from the Ladder Menu when Offline Online or Combined programming options are selected.

Only legal configurations are allowed. You are prompted to complete a configuration file for a specific style controller in order to have a file that matches the PLC you choose. You can specify the parameters by filling in the fields or simply by choosing from a predefined set of values in a popup menu. Saving a configuration is done through the Save function and its variants.

After the selection of configuration from the Segment Status Menu or Ladder menu Tools function, the following operations are available:

Utility PlcOps Overview I/O Map Ports Segments Loadable Cfg Ext Quit

The Overview entry is available in Offline **only** and the PlcOps is available in Online and Combined but **not** in Offline.

☞ **Note** Modsoft 2.4 included corrections to configuration size functionality. You may find that while using Modsoft 2.4 attempts to go on-line in the Combined mode without first transferring the program to the PLC (from Modsoft 2.4) can prevent access to the configuration section. If this situation should occur, use the transfer function to download the project to the PLC.

☞ **Note** Modsoft 2.5 prevents editing a PLC Configuration offline if the program contains a ConCept IEC loadable. A message is displayed:

OFFLINE EDIT OF CONFIGURATION IS NOT ALLOWED.

Concept IEC loadables are incompatible with Modsoft. However, Modsoft will allow upload and subsequent download (with warning) to the PLC programmed by Concept.

5.2 Utility

The Configuration Menu Utilityis similar to the Ladder Utility and different from the Main Menu entry. The difference is that the PLC & I/O Status function is available in an Online or Combined Configuration only. The Utility at this level offers both Reference Data and Symbol Table functions. These are defined in Chapter 8. The Offline pulldown with Quick keys appears as:

- Initial Utility Default Menu

```
Key Help
(PLC & I/O Status   if Online/Combined)
Program Information  [CTRL F1]
Dos                  [CTRL F7]
Quantum Backplane
```

- Program selected Utility entries

```
Key Help
(PLC & I/O Status   if Online/Combined)
Program Information  [CTRL F1]
Reference Data       [ALT F2]
Symbol Table         [ALT F1]
Dos                  [CTRL F7]
```

☞ **Note** No function keys are recognized when you have selected the DOS shell function.

5.3 Overview

The overview screen for the configuration file in the panel is always shown in the configuration screen. When the other editors of the configuration are called, they overlay the overview screen. There are different sections on the overview, which can be accessed via entries in the menu. Within these sections, the cursor can be moved freely. Pressing any key starts editing. Pressing <**Esc** > during editing in a field quits the entry and restores the old value. Each field has set ranges to define valid entries. Entries are checked with their ranges when you intend to leave this field or by pressing <**Ente** r>, <**Cursor Up**>, <**Cursor Down**>. Only if an entry is correct does the cursor leave the field. You can select from the following overview screen submenu selections:

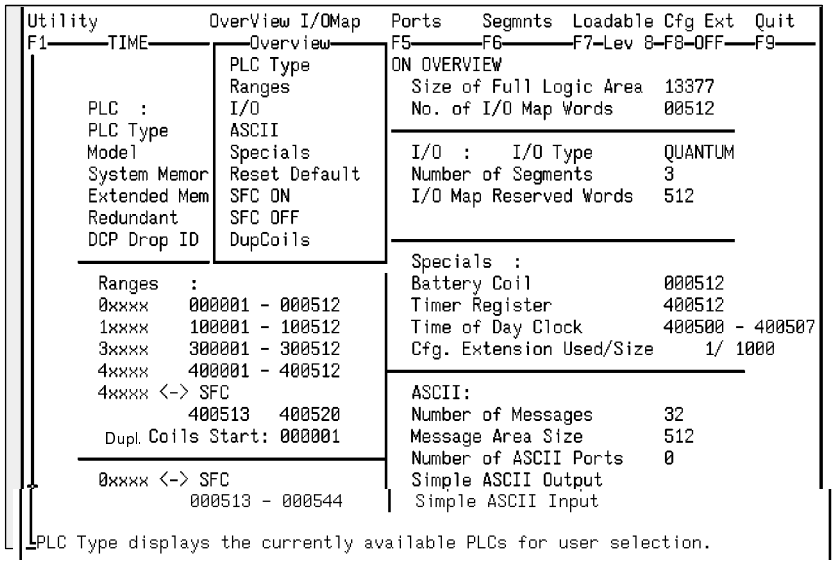


Figure 114 Overview Select Example and Menu Bar Selections

5.3.1 PLC Type

This part of the screen sets up the PLC model and memory configuration ***for a particular programmable controller model***. When starting up with a new system, this is automatically done (without the need to select "Overview PLC Type"), because some parameter settings and allowed functions within the Ladder Diagram Editor depend on the selected controller type. A warning is shown if you change items in the PLC Menu, after changing parameters in the configuration. Also there is a warning if parameters in the overview screen are changed before selecting the PLC type. After the selection of "PLC Type" from the PLC Selection List, you can specify Type and Memory parameters.

Dependingon controller type, the following may also be selectable:

```
Exec Pack or Model _______
System Memory      ______
Extended Memory    _______
Redundant          _______
DCP Drop ID        _______
```

You can change data by first calling the appropriate function and then selecting the appropriate parameter from a selection list. Only values listed in the selection list can be selected. Changing the PLC type to a controller without ASCII causes all ASCII entries to be set to zero. No other entries are initialized when changing the type, but when leaving the configuration, a check for consistency is done.

5.3.1.1 PLC Type

"Type" sets the controller type as a top level classification. This list will be updated as other controllers are added. You select the desired type using cursor up or down keys. In the example a Quantum is being selected to change the initialized default configuration on the distribution disk.

Both the S908 (800 Series I/O) and S901 (200 Series I/O) I/O-Processors are supported.

Type selection of the 68X or 78X series, including E and L models, produces another select prompt for either a

512 S908 or
1K S908

These values are related to the S908 upgrade that supports 16 drops of 1024I/O points vs 32 drops of 512 I/O points. When you pick the 1K S908 the "I/O Type 800" data field is altered to indicate the selection by adding a **-1K** to the field.

Note The process of changing PLC type resets the Duplicate coils field, In the configuration Overview display, to zero. This is because it is potentially dangerous to re-use coils and the user may forget that this field is set in the new PLC application.

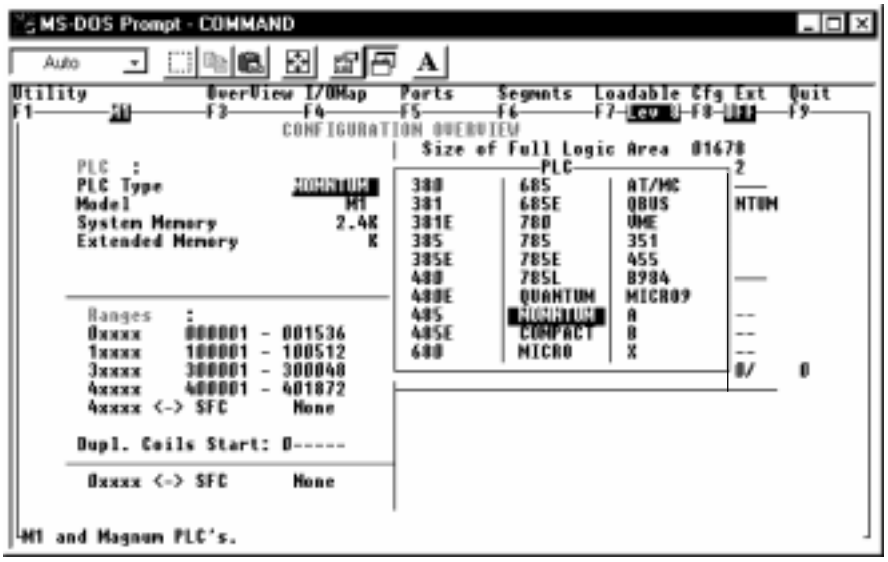


Figure 115 Example of controller types in Version 2.5 Release

Expert If you choose the Modicon **Micro** as a controller type you will find an architectural, and conventional difference in operation and terminology. To help you configure this controller family you may want to reference the information in ***Appendix G.***

5.3.1.2 Model

This field heading replaces the `Exec Pack` heading for those controllers that do not have separately defined parts, such as the Quantum and Micro Product family model selections for example.

5.3.1.3 Ladder Logic Memory

"`Ladder Logic Memory`" allows selection from the available Controller type memory configurations. The valid memory configurations available depend on the PLC type:

Controller Configuration Selections

984 Controller Types	Memory
984-A	16k or 32k
984-B	32k or 64 k
984-X	16k or 32k
AT/MC	12K, 16k or 32K
AT4	32k
QBUS	12k
VME	12k
VME924	32k
VME424	64k
984-100/102	4k
Micro9	4k
B984	4k
984-380	1.5 k, 4 k or 6 k
984-381	1.5 k, 4 k or 6 k
E984-381	1.5 k, 4 k or 6 k
984-385	1.5 k, 4 k or 6 k
E984-385	8k
D984-385	8k
984-455	8k
984-480	4k or 8k
E984-480	8k
984-485	4k or 8k
E984-485	8k
984-680	8k or 16k
984-685	8k or 16k
E984-685	16k
984-780	16k or 32k
984-785	16k or 32k
E984-785	32k/64k 0r 48k/32k
L984-785	32k/32k - 0 k Extended Memory 48k/32k - 26 k Extended 16k/64k - 74 k Extended 32k/64k - 98 k Extended

Momentum Controller Type	Memory
Momentum-M1	2.4k, 12k, 18k
Momentum- Magnum	10k or 18k

Quatum Controller Type	Memory
Quantum x13-0x	8,16,32 or 48K Ver 2 or higher
424 0x	64k-96k extended Ver 2 or higher
x13 0x-x	8,16,32 or 48K Below Ver 2
424 0x_x	64k-96k extended Below Ver 2
434 / 534	64k-96k extended

CompactController Type	Memory
984-120, A120	1.5k or 4k
984-130, A13X	4k
984-A141	8k
984-145, A145	8k
984-241/245	8k
984-251/255	16k - 24k extended

Micro Controller Type	Memory
311/00,01,02.03	2k
411/00,01,02,03	2k
512/00,01,02,03	3.1k
612/00,03,04	3.1K or 8k for 04

5.3.1.4 Extended Memory

This function establishes the presence of an extended memory in the configuration file. Valid extended memory configurations for 984B are 0k, 32k, 64k or 96k and for 984-785L and 785E are 0k, 26k, 74k, 98k. The 984B, 785L and 785E provide the facility to define extended memory. The E984-785 has 96k or 24k extended depending on your configuration selection for User Memory/State RAM partition. Quantum Series offers up to 96k extended memory, the assignment of which is based only on picking the controller type. Compact E251/E255 has 24k of Extended Memory the value of which is entered in the display when the Model is selected.

5.3.1.5 Redundant

If you choose to use the programmable controller as a redundant system, the configuration must be set for this purpose. If the Controller Type does not support redundancy this field is not displayed.

5.3.1.6 Model (Previously calledExec)

When a 984-8 type controller is selected, an assumed Exec type is listed. You can edit the Model to provide the proper function if the default is incorrect. If a controller in the 984 A/B/X family is selected, the Exec field is cleared.

Because the machine code of the Loadables is totally different for series 8x modules, it is necessary to have different library files for the 8x controllers.

Users have the option to select which 984-X8X Executive Cartridge to configure. The options are:

EXXX-902
EXXX-904
EXXX-914

or the fixed Model E executive designated as a 924 or QNT

These Executives must be selected in order to produce the correct configuration for the 984-X8X. Check the Cartridge label and make sure you are using the correct Executive.

☞ **Note** The 984-X85 controllers, except the -685, automatically default to the EXXX-904 Executive Cartridge.

If the Controller is a 984-X85 or 984-145, the MSTR block will replace the CKSM DX in the E904 Executive Cartridge.

☞ **Note** In Model E redesigned slot mount controllers you have no removable executive or memory cartridges, the executive characteristics are read by Modsoft and posted to the overview display.

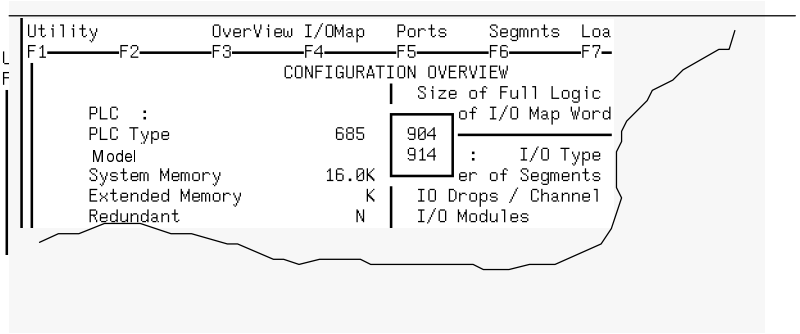


Figure 116 Example Of Executive ID Selection

The above Executive list offers you a choice of PC-0984-685 implementations where the 904 and 914 are the standard and enhanced versions. If you have a Model “E” controller the Executive type 924 or QNT presented for selection. The Model “E” family executive ID’s appear in the Exec Pack or Model field for each controller type in the PLC Overview Configuration screen.

☞ **Note** If you have a controller type selected and you are not familiar with the Model (executive) selection list, refer to the Systems Manual or User Guide for the specific Controller.

Like the 984 E Model, the Quantum Series Controller Executive is downloaded to EEPROM and is not a removable part. When you select the Quantum PLC type, the `Model` field displays the `Model` that you select. As listed in the configuration table the Models

(Model) x13-0x; 424-0x support Equation Networks and fast I/O blocks
(Model) x13-0x-x 424-0x-x do NOT support Equation Networks or fast I/O blocks

5.3.1.7 DCP Drop ID

If you select a controller that supports the D908 (Distributed Control Processor), the area that is labelled DCP Drop ID will allow entry of a D908 drop number.

5.3.2 Ranges

This part of the screen is used to set the maximum number of references for each type. They are needed as a range check for offline logic programming.

0XXXX References. Your chosen range of defined logic coils can be entered by an upperbound. The total number must be a multiple of 16. It is automatically adjusted up to the next multiple of 16. On the initial configuration the default value is 1536. Additionally, 32 0XXXX references are defined for use in SFC (if enabled). The SFC number of coils cannot be edited. Range checks are made including the number of SFC 0XXXX references. The number of 0XXXX references is also checked with the available memory on the Controller. One word of reference for each block of 1 to 16 (one bit per discrete) entries.

1XXXX References. The range of defined discrete inputs available for the program is entered as an upper limit. The total number must be a multiple of 16. The default value is 512.

3XXXX Registers. The range of defined input registers available for the program is entered as an upper limit. The default value is 48.

4XXXX Registers. The range of defined output/holding registers available for the program can be entered. On the initial configuration the default total number is 1872.

To determine the total number of registers available for internal use when SFC code is generated, you input the upperbound of 4XXXX registers; the system then calculates the corresponding additional area. Starting with the first 4XXXX register following the range you have defined for standard use, Modsoft adds 1 register. The appended registers increase by one 4XXXX register for every used SFC row (a requirement for the code generator). If you transfer a configuration with less than two 4XXXX register or less than 48 0XXXX references, a problem is created within SFC structure. In this case a warning is displayed and the ranges are automatically expanded. If the configured range limits are exceeded by this expansion, an "Illegal Range" error message is displayed when Modsoft leaves the Configuration Editor and you will need to assign a higher available range of registers.

Note: The logical memory arrangement of the reference registers are discussed in the Modicon Ladder Logic Block Library
840 USE 101 00

Duplicate Coils Start: The default value of the First Reusable Coil is zero. Changing this value offline requires that the user has set the User Preference File (`Ladder` Pulldown) to allow reusable coils. (or start modsoft with the /d switch). In the absence of the UPF entry (or /d switch) the `Overview` field Reusable Coils is greyed and the value not changeable. Once you have set the value and gone back online you can only reuse coils from the point set in the configuration (even though you might have had some reusable under that value in the original Program).

This field, as with all other overview fields, is not changeable online.

Tip In non-Quantum modules the maximum I/O limit is 16383 for both 3XXXX and 4XXXX registers (14bit effective address)

Note If SFC registers and coils are not required then Modsoft can be appropriately configured with SetSFC (On-Off toggle) from the Overview menu.

Note Programming modes provide an “Over range ” message, if you enter a reference beyond the stated range. The Reference Data Editor allows you to enter the reference without an error message but there is no value associated with that reference.

Expert: With Momentum M1 configured
0x + 1x sum to 2048 maximum
3x + 4x sum to 2048 mazimum

5.3.3 I/O

The Memory information “Size of Full Logic Area” and “Number of I/O Map Words” depend on the parameters that are specified in the configuration function.

The next screen partition varies depending on the selected controller Type, but generally has:

I/O Type - Number of Segments
I/O Drops - I/O Modules

5.3.3.1 I/OType

The I/Otype, and some sub fields, are set for you ***dependingon the PLC selected*** Either 200 or 800 series I/O for the 984 A/B/X; 800 series or Quantum Series for Quantum are the types displayed. If the controller has a S901 board, you must select 200. If the I/O board is an S908, then you must select 800 as the type.

The 984A and 984B controllers support either 200 or 800 series I/O.
984-X8X and 984X series controllers support 800 series.
The Compact and Micro support A120 I/O.
The Micro9 supports 300 Series I/O.

The 984-351/455 supports 500 Series I/O
Quantum supports Quantum Series I/O and/or 800 Series I/O
Momentum Supports Momentum I/O Base and I/O Bus Modules

5.3.3.2 Number of Segments

This entry allows you to set the number of segments. The default number of segments displayed is 32 but the actual value will vary with the selected PLC type.

For 800 series I/O the number of segments cannot be less than the number of drops entered. The 200 series I/O segment numbers is always *equal* to the number of channel *pairs.*

5.3.3.3 I/O Map Reserved Words

This field (if present) has an initial default value of 512. This value is editable and represents the quantity of memory allocated for I/O register Mapping.

5.3.3.4 I/O Drops/Channel Pairs

The number of I/O Drops/Channels and I/O modules affects the size of the I/O Map. I/O Drops lets you set the number of available I/O drops. Default value is 1 drop. The maximum number depends on the selected Controller type.

5.3.3.5 I/O Modules

The number of I/O modules affects the size of the I/O Map. Total Number of I/O Modules lets you set the number of available I/O modules. For each module defined, 6 words are allocated as a worst case definition, so later changes can not increment the size of the I/O Map.

The maximum number of I/O modules depends on the number of I/O drops; 32 modules (984x, 984-680, 685 and 984-780,785) per remote drop maximum, 21 modules per local drop maximum (984-380, 381, 385 and 984-480, 485).

The size of the currently defined I/O Map and remaining memory words are displayed on the top of the overview screen. This information cannot be edited.

Compact 984's support a maximum of 18 Modules.

5.3.4 ASCII

The ASCII screen option lets you set the limits for the ASCII functions. The following parameters can be configured:

- Number of Messages

- Message Area Size
- Number of ASCII Ports
- Simple ASCII Output
- Simple ASCII Input

The 984-351, -380, -381, -385, Micro9, -100, -102, -120, -A120 -130, -A13x, -A141, -145, -A145 and Micro have no ASCII capability.

5.3.4.1 Number of Messages

This entry serves to enter the number of ASCII messages. The maximum is 9999 messages for 984 A, B and X controllers. The maximum number of messages for the Quantum, 984-680/5, 985-780,785 and 984-480/5 is 999. The value entered reflects the total number of messages and not the 984 message number. On the initial configuration the default value is 0.

5.3.4.2 Message Area Size

This entry serves to set the total number of words that is reserved by the controller for ASCII messages. The default value is 0. One word of memory equals two ASCII characters. For example, a message area of at least ten words is needed to adequately cover a message length of seventeen characters due to a three character overhead per message.

5.3.4.3 Number of ASCII Ports

This entry serves to enter the number of RS-232-C ports included in the system for ASCII communication. This value can be a maximum of 32. The default value is 0, i.e., no ASCII port defined. The number of ASCII ports available depends on the number of drops defined on the selected Controller type. The entry is checked with the number of drops when leaving the configuration. When editing the two values above, a check is done to determine if there is enough memory in the Controller. ASCII communications parameters at default are; 2400 baud, Even Parity, 7 data bits and 1 stop bit.

5.3.4.4 Simple ASCII Range for Input/Output

These entries serve to enter 4XXXX holding registers that indicate the first register of a group of 32 registers reserved for Simple ASCII input and output. The first register is also the control register. Depending on the selected Controller, different Simple ASCII ports are available. The 984-B has a Simple ASCII input and Simple ASCII output. 984-A has Simple ASCII output. No other controllers support simple ASCII. All 32 4XXXX registers have to be within the defined range of 4XXXX in the overview.

5.3.5 Special

Various special options can be configured:

5.3.5.1 Battery - Timer - Time of Day

The `Battery coil` selection lets you enter a 0XXXX reference number. This coil reflects the status of the battery backup system. The Configuration default value is a blank field meaning no battery coil is assigned. When the contacts (of the coil you enter) are used in a program they respond to a *coil energized* condition to indicate a low battery voltage.

The `Timer` register selection lets you enter a 4XXXX holding register number the contents of which increment each 10 milliseconds of clock cycle. The Configuration default value is a blank field meaning no timer register is assigned. The register you select is available in your program and has a free running value that ranges from 0000 to FFFF Hex (wraparound to 0000).

The `Time Of Day` (TOD) Clock is controlled by the first register of a series of eight consecutive 4XXXX registers, reserved for the time of day clock. All eight 4XXXX registers have to be within the range of 4XXXX registers defined in the configuration overview.

The TOD data input has an automated display menu called SetHardwarClockfound under "PlcOps" in the Online/Combined function selection. However, Modsoft is compatible with other Modicon panel products and you can also update the TOD clock given a completed configuration, your knowledge of what 4XXXX registers are defined, the ability to use a reference editor and the following (USA) data format:

4XXXX	The control register	
	Bit 1 (MSB) 1= set clock values	-You must set these bits in
	Bit 2 1= read clock values	-Logic (or manually) to set the desired condition
	Bit 3 1= done bit	
	Bit 4 1= error bit	
4XXXX + 1	Day Of the Week (1 - 7)	
4XXXX + 2	Month (1 - 12)	
4XXXX + 3	Day (1 - 31)	
4XXXX + 4	Year (00 - 99)	
4XXXX + 5	Hour (Military) (0 - 23)	
4XXXX + 6	Minutes (0 - 59)	
4XXXX + 7	Seconds (0 - 59)	

The reference display and a set of example values for the Time Of Day clock are illustrated. By using binary on the first register you can see the set bits directly. The clock “set” bit is cleared automatically once the clock is set. When the read bit is set it remains set until you or the program clear it. Keeping this bit on provides continuous updating of the registers. The done bit is set each scan in which set or read are completed and there are no errors. The error bit indicates a bad value in the table.

`Configuration Extension Size`
The Specials parameter box is where you can see how much memory has been included under the Configuration Extension configuration. This field is posted for information only and can not be altered from the Overview screen. Refer to the Configuration Extension description later in this chapter for setting the size.

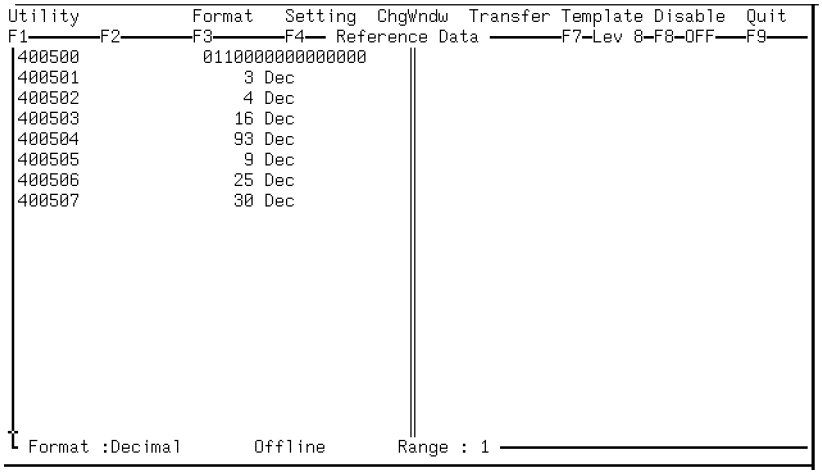


Figure 117 Time Of Day Clock Register Format Example

This is an example of Tuesday April 16, 1993 at 9:25 and 30 seconds.

5.3.6 Reset Default

This selection allows you to return to the start up default environment with all known values for the PLC Type selected. The distribution disk default PLC Type is a 984-385.

5.3.6.1 Set SFC

The Overview pulldown has two choices displayed that toggle between SFC ON or OFF. When ON, SFC and Macros are allowed and extra registers are automatically assigned. When

OFF, no SFC overhead exists for coils and registers nor are there any SFC Steps or segments displayed or documented. Only Absolute Network numbers are displayed.

Note The number of 0x and 4x registers are incremented (0x are incremented by 16 and 4x are incremented by 1) to account for SFC. This can result in “Illegal Range” error when the consistency check is performed upon exiting the configuration session.

5.4 I/O Map

The I/O Map (formerly called a Traffic Cop) is used to direct the flow of data between the various I/O modules and the logic program. It is the tie between the references used in the logic program and the I/O module connection points.

5.4.1 200 Series I/O configuration

When 200 series interface is selected, via Configuration I/O Type (S901) at the Config OverView screen, the panel displays the appropriate 200 series I/O Map screen. (A detailed reference for 200 series interface is “J290/J291 Remote I/O Retrofit Installation and Operation Guide” Schneider Automation, Inc. Document Number GM-J290-001 Rev A). The number of slots per channel is 8 and there are 2 channels per drop (2 Input and 2 Output). The even numbered channels per drop are Slots 401-408 inputs and 501-508 outputs, and the odd numbered channels are Slots 201-208 inputs and 301-308 outputs. In order to use 200 series I/O, the controller must have a S901 I/O card in the controller housing (An S908 can be used with J290/291 interface units). Both the 984A-200 and 984-B-200 controllers have 32 channels, 16 channels In and 16 channels Out.

Entries to the 200 series I/O Map are made by typing reference numbers (0X or 4X for Output channels and 1X or 3X for Input channels). Even and odd channels each have eight input and eight output slots. You can toggle between the Even and Odd channels by pressing the <**PgUp** > and <**PgDn**> keys. The 200 series I/O Map is displayed in the next illustration.

```
Utility          ClrDrop  HoldTme  ASCPort  Drop                    Quit
F1-------F2-------F3-------F4-------F5-------F6-------F7-Lev 8-F8-OFF--F9----

                                  I/O MAP
                               200 SERIES I/O
     Drop             :  1                Channel    :     ODD,   INPUTS
     Drop Hold Up Time :  3     (x100ms)  ASCII Port :     0
    ======================================================================

     Slot            Input                Output

      201            ------               ------
      202            ------               ------
      203            ------               ------
      204            ------               ------
      205            ------               ------
      206            ------               ------
      207            ------               ------
      208            ------               ------

 Hold Up Time changes the drop's signal state time.
```

Figure 118 200 Series I/O Map screen

Edit functions are available from the menu line and provide the ability to Delete a Drop or to Get a Drop. The `Delete Drop` function will delete all modules in all racks of the current drop. The GetDropselection has a pulldown that lets you select the previous or next drop or offers you an entry to request the drop by number or to move between Next and Previous Racks. The ASCPort selection lets you assign the proper physical port to the drop/slot data.

5.4.1.1 800 Series Configuration

The 800 Series I/O Map lets you match the 984 controller I/O addresses with what will be installed or with what actually exists in the field. The I/O Map also tells the controller how to use an input signal in user logic and where to send an output signal. The format of register data (BCD -binary coded decimal or BIN binary) is specified on this screen. The screen objective is to load the card selections and reference number selections to complete the configuration of the I/O system.

In the Configuration Overview, you must specify the number of I/O drops and I/O modules. These numbers are checked when entries to the I/O Map are made. Each drop consists of two or five racks depending on the Controller type. Functions are available to erase a single slot from the screen and to delete an entire drop of I/O settings (after confirmation and not while running). If you try to use a reference more than once a warning is displayed.

Paging through different racks is done with <**PgUp**>, <**PgDn**> keys. Menu entries are available to get the next drop, the previous drop, or a drop with a specified number. Each rack can have up to 11 slots; a table is displayed with 11 lines to specify the modules of this drop. The numbers in the first column indicate the physical location of a module for a specific rack. Modsoft Runtime files contain descriptions of the different kinds of modules. GCNFTCOP.SYS includes the description of ”B8” modules, with module number, number of inputs, outputs description and Analog identifier .

If there are any new I/O modules, you can add them to the GCNFTCOP.SYS file in the RUNTIME directory using any text editor. The current list is found in this file, which has the following format:

If the first character on a line is a semicolon, (;) the text that follows is a comment.
An I/O module I/O Map definition line starts with the module name followed
by a **comma.**
The next parameter is the Schneider Automation Module ID followed by a comma (,).
Next is 0 or 1 followed by a comma (,) where:
0 = Other modules may be inserted in this drop.
1 = This is the only module that can be inserted in this drop.
Next is the number of input bytes followed by a comma (,)
Next is the number of output bytes followed by a comma (,)
Next is the Module description (up to 19 characters) followed by a comma (,)

Last is the Analog Module Identifier Where:

0 = A discrete module (may take 0x, 1x, 3x, or 4x references)

1 = Analog module (may take only 3x, or 4x references)

```
; EXAMPLE
; This is a comment line
;
B804,04,0,0,2,115 VAC 16-OUT       ,0,
```

Modsoft Runtime data is available to list the four types of modules:

- Modules for controller and power supply (combined)
- Power supply modules
- Responder modules
- Special option modules.

800 Series I/O Modules Supported (as of Modsoft 2.2)

Unit Designation	Type
B802	8-OUT
B803	8-IN
B804	16-OUT
B805	16-IN
B806	32-OUT
B807	32-IN
B808	16-OUT
B809	16-IN
B810	8-OUT Isolated
B814	8-OUT Relay
B816	16-OUT Isolated
B817	16-IN Isolated
B818	32 Discrete Output
B819	32 Point Input
B820	8-OUT
B821	8-IN
B824	16-OUT True High
B825	16-IN True High
B826	32-OUT True High

Unit Designation	Type
B827	32-IN True High
B828	16-OUT VTTL
B829	16-IN VTTL
B832	16-OUT True Low
B833	16-IN True Low
B834	8-OUT
B835	8-IN
B836	16-OUT
B837	16-IN
B838	32-OUT
B840	Reed Relay NO
B842	Reed Relay NC
B846	Analog Mux
B849	48V AC/DC IN
B853	125V DC IN
B855	16-IN SAFE
B862	REG 4 CH OUT
B863	REG 4 CH IN
B864	REG 8 CH OUT
B865	REG 8 CH IN
B868	REG MUX OUT
B869	REG MUX IN
B872	ANALOG 4 CH OUT
B873	ANALOG 4 CH IN
B875	ANALOG 8 CH IN
B877	ANALOG 16 CH IN
B881	BIDIR 1 REG
B882	BIDIR 2 REG
B883	BIDIR 3 REG
B884	BIDIR 4 REG
B885	BIDIR 6 REG
B886	BIDIR 8 REG
B887	BIDIR 12 Reg
B888	BIDIR 16 REG
* 410	1 AXIS SERVO
* 3220	4 AXIS SERVO
* 3240	8 AXIS SERVO

Unit Designation	Type
S908	RIO Processor
S911	Hot Standby
D908	Distributed Control Processor

* See the Paragraph “Special I/O Map Entries”

To display the 800 I/O list, press **?**, and a short description is placed on the last line for each Module on which you place the cursor.

Module Definitions For each slot you can specify:

Module Type - Reference Numbers - Mode

In the Module Type column you can enter or edit a valid B800 series module, as defined in the description file mentioned above. There are two methods of adding modules. The first is by typing the module name. The second is by pressing ? and selecting the module name from the I/O list on the display. Leaving this field will cause the module description to be displayed in the last column of the current line and the cursor is set to the input/output field (depending on which type of module). Once a module is entered, you cannot leave a line, or leave a rack or drop if no inputs or outputs are defined. If no modules are entered you can leave the drop.

Reference Numbers for Input and Output are the next entries. The number must be a 1XXXX or 3XXXX number for an input module and a 0XXXX or 4XXXX for an output module. In general, 0XXXX and 1XXXX references must start at 16bit word boundaries, but some 800 modules allow programming in steps of eight bits. Because the 800 series address is 14 bits long you are limited to the highest register used of 16383.

Note Remember that some modules have **both** input and output references.

You can not define both registers (3X/4X) and discretes (0X/1X) for the same bi-directional module. There are limits for the maximum number of I/O references that can be accessed depending on the currently selected Controller type. For register references, you can choose between Data types including Binary and BCD mode. Discrete references are assigned binary mode only. The 800 series I/O Map should appear:

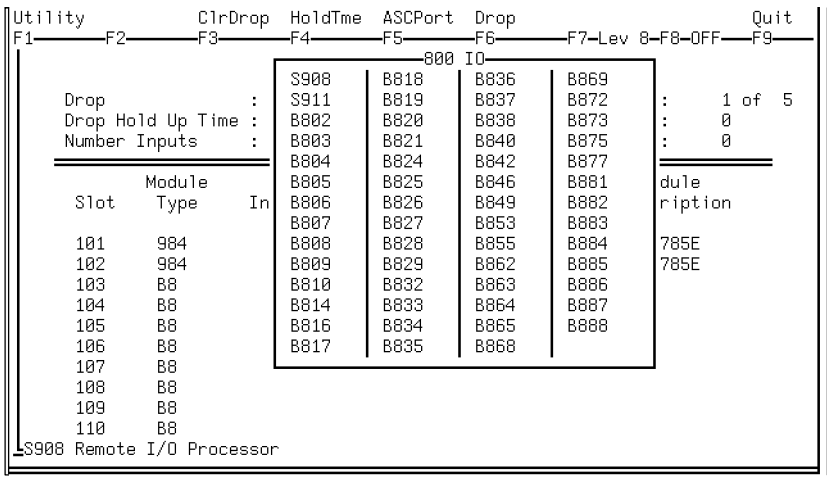


Figure 119 I/O Selection List Example

By using the selection list, the module and it's parameters are displayed in the I/O Map without additional keystrokes. Some sample entries are illustrated below.

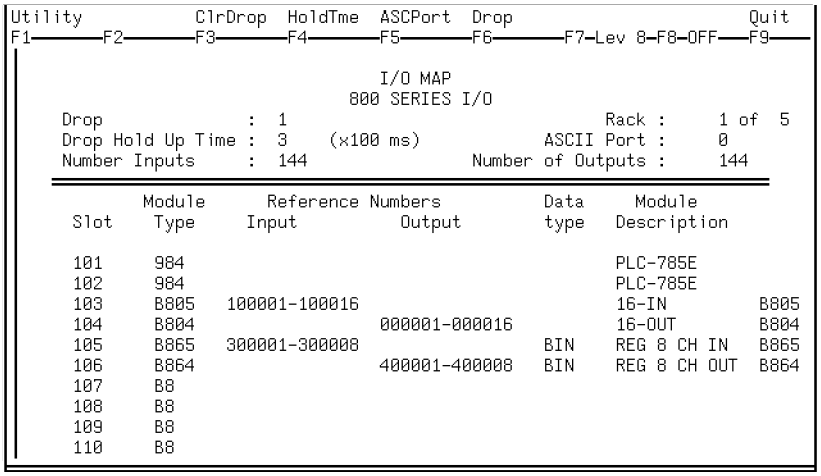


Figure 120 800 Series I/O Map with Some Module Data Entered

Special I/O Map Entrie s Special Modules, such as, J290/J291, D908 and the motion modules 410, 3240, 3220 are included in the 800 I/O list. To use them you must choose a remote I/O drop and place the entry in the ***first*** slot of rack 1.

- D908

A different procedure in defining the references for DCP is provided here. In the following D908 slots, blocks of registers can be defined instead of modules. In the first slot the number of registers in the block is defined. In the next column the first register (or reference) is defined and Modsoft calculates and displays the last number. The total number of registers for the blocks defined in the description file is checked against the configuration. The DCP entries are contiguous, .i.e., every block entry must follow the next, there can be no empty block row. An error is displayed if you try to enter a value two slots below the last.

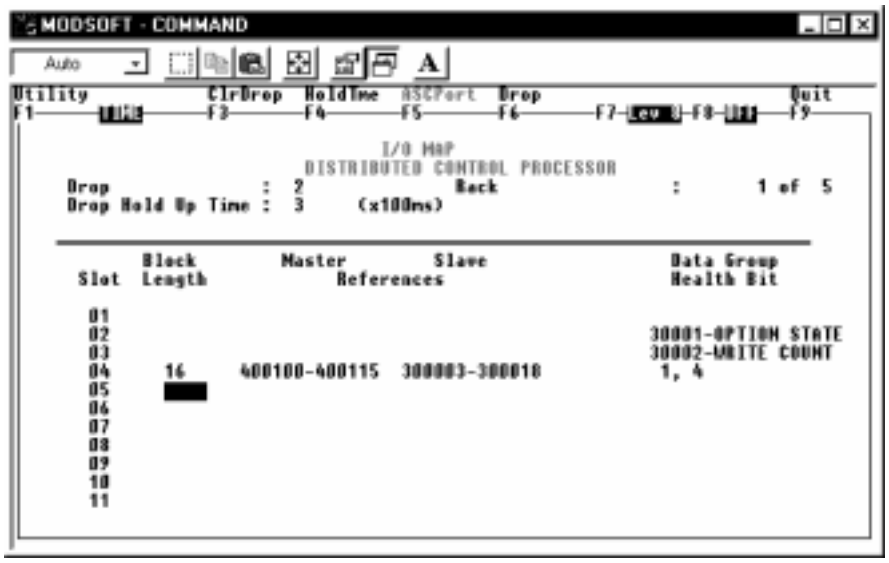


Figure 121 D908 Distributed Control Processor I/O Map

- J290/291

For the J290/1 the even numbered racks are used for input and the odd numbered racks are used for output. There is no rack number # 1 for this special drop. The initial screen for these I/O Map entries illustrates the fact that they are captioned under 200 series I/O.

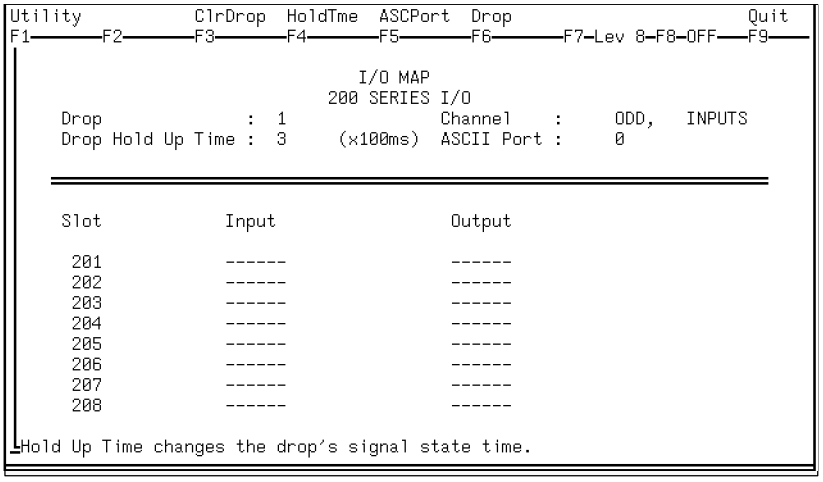


Figure 122 J290 - J291 I/O Map Initial Screen

Drop Parameters On the top of the screen, there is information on the current drop, the current rack and the used inputs and outputs. Also on the upper part of the screen you can specify two parameters affecting the current drop:

Drop Holdup Time and ASCII Port

Holdup Time defines the amount of time the drop will scan its input and output if communication from 984 has been lost. The value can be any integer between 3 and 65535. The number entered is multiplied by 100 milliseconds. The default is 3, which equals 300ms.

5.4.1.2 ASCII Port

The ASCII port number can be assigned to be any **odd** number between 0 (for none) and 31, but only within the range of ASCII ports defined in the configuration overview screen.

5.4.1.3 984-230 (Micro9) I/O Map

984-230 (Micro9) I/O Map Since the 984-230/234 (Micro984) is a replacement for the Micro84, this controller uses 300 series I/O modules for I/O processing. Upon selecting the Micro9 controller in the configuration, the software automatically displays a 300 series I/O Map screen. Upon selecting I/O Map Editor, Thus:

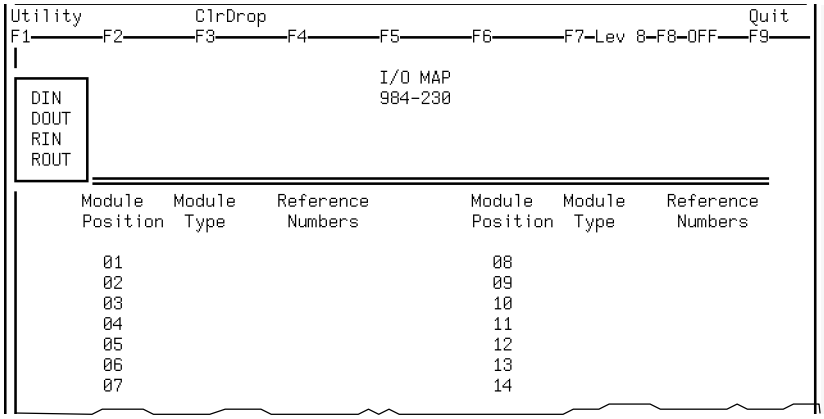


Figure 123 984-230/234 I/O Map for the Micro 9

There are four module types recognized by the 984-230/234 I/O Map and selectable from a pulldown list as illustrated above:

- discrete input module (DIN) - discrete output module (DOUT)
- register input module (RIN) - register output module (ROUT)

The discrete modules are 8 bit and the register modules are 32 bit, or two registers. A 1X or 3X start reference can be assigned to either input modules and 0X or 4X can be assigned to either output modules. The I/O Map automatically calculates the range of references that will be assigned to the modules. If the module is 8 bit and the reference used is 0X or 1X, the starting reference must be a multiple of 8 plus 1. If the module is 32 bit and the references used are 0X or 1X, the starting reference must be a multiple of 16 plus 1. If the module is 32 bit and the references used are 3X or 4X, the I/O Map assigns two registers. The maximum number of I/O modules allowed per module type:

- discrete input 8 modules - discrete output 8 modules
- register input 2 modules - register output 4 modules.

A maximum of 14 modules are permitted. No empty slots are allowed within a group of modules due to the hardware constraints. Delete Drop lets you completely erase the entire drop of entries.

5.4.1.4 *984-120 I/O Map*

Upon selecting the COMPACT type you are supplied with a model list to select from and you are automatically assigned a unique I/O Map screen.

The **Type** List includes:

0120	0130	0145	A120	A13X	A141
A145	E241	E245	E251	E255	

The following is an example of this I/O Map screen.

☞ **Note** Some I/O modules require the loading of a specific driver. The drivers for #MOT, ADU214, ADU216, DSC1 and SVI are found in the path C:\MODSOFT\A120IOD. Use the same loadable sequence as other loadable products supplied on floppy disk media, just use the above path instead of “A: filename” (see Paragraph 5.14).

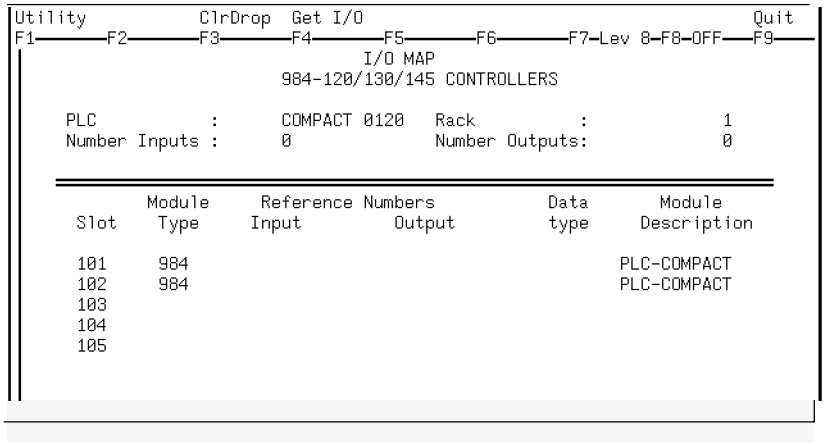


Figure 124 984-120 Model I/O Map

There is a maximum of 5 modules per rack and a maximum of four racks are available on the 984-120. In the first rack, the Controller takes up the first two slots. The I/O modules currently supported are listed in table 2 and can be displayed by pressing the ? key while the cursor is on the Module Type field. To get to racks 2,3 or 4 use the <**PgUp**> or <**PgDn**> keys.

Currently Available 984-120 I/O

I/O Map Entry	Module Description
DEP208	8 IN 230VAC
DEP216	16 IN 24V
DEP220	16 IN 24V

DEP209	8 IN 115 VAC
DEP210	8 IN 115VAC
DEP211	8 IN 110VAC
DEP214	16 In 10-60 VDC
DEP215	16 IN 5VDC
DEP217	16 IN 24VDC
DEP218	16 IN 115VAC
DEO216	16 IN 24 V
DAO216	16 Out 24 V
DAP204	4 OUT 24 V
DAP208	8 OUT 24 V
DAP209	8 OUT 115 VAC
DAP210	8 OUT 115VAC
DAP212	8 IN DC 4 OUT 24 V
DAP216	16 OUT 24 V
DAP217	16 OUT 24VDC
DAP218	16 OUT 115VAC
DAP220	8 IN 8 OUT 24 V
ADU204	4 Channel A/D +/- 0.5V
ADU205	4 Channel A/D +/- 10V
ADU206	4 Channel A/D
ADU21x	8 Channel Multirange Analog In Opto-Isolated
ADU214	4 Channel RTD Input
ADU216	8 Channel Analog In Opto-Isolated
DAU202	2 Channel D/A
DAU204	4-Channel Analog Out Opto Isolated
DAU208	2 Channel D/A
ZAE201	Position / HS Count
ZAE204	High Speed Counter
MOT201	1-Axis Motion
MOT202	1-Axis Motion
VIC2xx	4 Channel VCR Pulse Input

☞ **Note:** When Mapping an 8 piont module **you nust use a 16 point** reference boundary. For example mapping a dap-212 followed by a dap-208 the PLC will not start and a 4000 error exists. Shift the reference to a 16 bit boundry and it will function properly.

The format of this runtime file (GCNFA120.SYS) is the same as described in the 800 series I/O paragraph).

```
; EXAMPLE
; This is a comment line
;
DEP216,12,0,2,0,16-IN 24V   , 0
```

5.4.1.5 *B984-100 / 102 I/O Map*

The 984-100 I/O Map has preassigned I/O modules and associated reference numbers. The 984-100 is also called the B984 High Speed Logic Solve Module and is used like an I/O module. The I/O Map appears as:

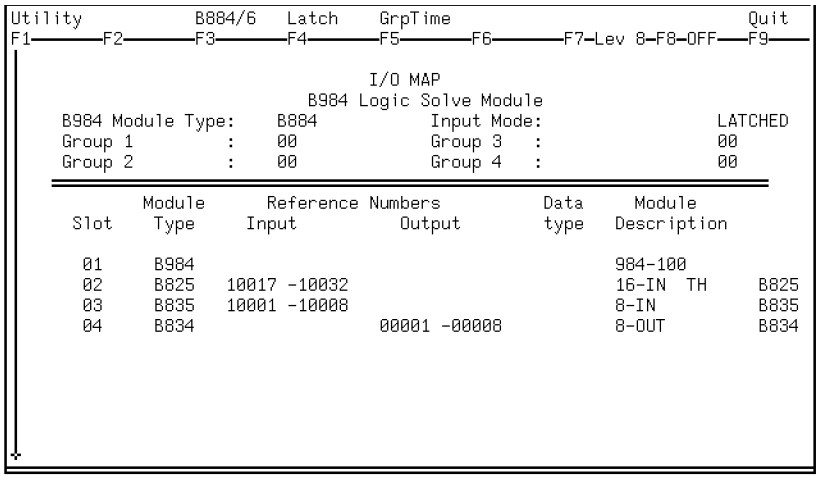


Figure 125 984-100 I/O Map for High Speed Logic Module

The Menu selections to support this PLC type allow you to select from either a B884 or B886 I/O Module type, both of which are listed when you select the "B884/6" menu entry. You can set the input signal conditions to Latched or Unlatched using the "Latch" Menu entry. The "GrpTime" allows you to go through each of the four groups and set a time value between 0 and 99.

5.4.1.6 500 Series

The 984-351 and 455 controllers have a special I/O Map for 500 series I/O but will be displayed as 800 series. The 984-455 has a maximum of seven drops
(2 local 5 remote). The two local (500 I/O) drops have four housings preassigned. Housing two has B817 modules assigned, housing three has B816 assigned and housings four and

five has B881 modules assigned. Housing one does not exist. The five remote drops are for 800 I/O use.

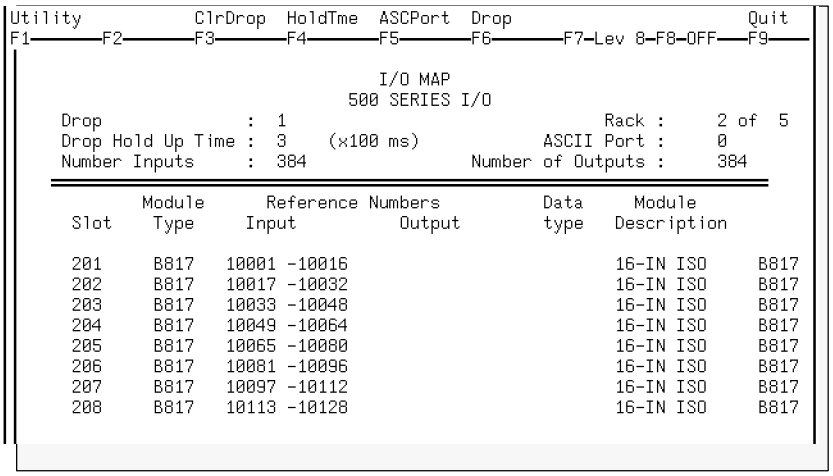


Figure 126 984-455 Controller I/O Map

5.4.1.7 984-Motion Series

If you are planning to use motion modules, an empty remote drop is required. You will see “Slot 101 with B8- -” at which point, just type the 410,3240 or 3220 as the entry of a remote drop type (i.e., at the first “Module Type” slot.). You may also select from the list.

The 1 Axis (410) I/O Map screen is illustrated. To make this module entry, type the 410 in the Module Type field or select the appropriate Motion module from the select list displayed by pressing the ? key with the cursor on the module type field.

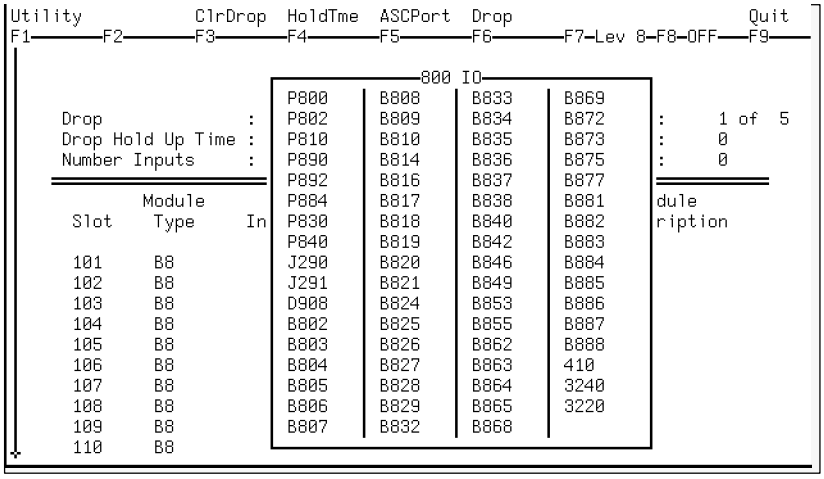


Figure 127 Special I/O Module Select List

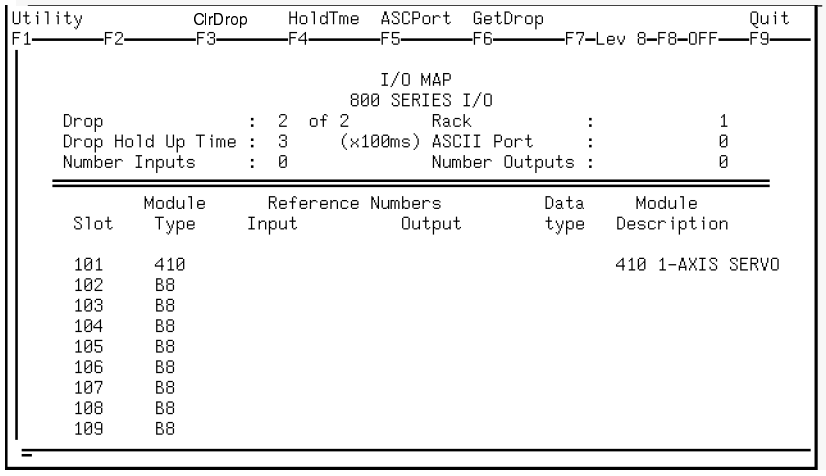


Figure 128 Motion Control I/O Maps

5.5 I/O Map (Micro)

The I/O Map lets you match the Micro controller I/O addresses with what will be installed or with what actually exists in the PLC. It also tells the controller how to use an input signal in user logic and where to send an output signal. The format of register data (BCD which stands for **b**inary **c**oded **d**ecimal or BIN which stands for **bin**ary) is specified on this screen. The screen objective is to load the IO selections and reference number selections to complete the configuration of the I/O system. In the configuration overview, you must specify the number of I/O modules. These numbers are checked when entries to the I/O Map are made. If you use a reference more than once a warning is displayed.

The Micro includes a number of prompt lists and error messages. The I/O Map is provided in three different screen overlays that correspond to the Operating mode you have chosen.

The first screen is for a Micro in Single Mode
The second screen is for a Micro in Parent Mode
The third screen is for a Micro in Child Mode

The examples that follow show each of the three Mode selections mapped with the defaults and types of Lists and typical entries you might make. You get a screen upon initial selection of the Configuration Menu Single selection. The entry I/O Map screen for a configured parent is illustrated below. The illustrated data is the default condition. The next illustration is of the Child selection.

```
Utility          Del I/O  HoldTme  Get I/O                              Quit
F1———F2———F3———F4———F5———F6———F7—Lev 8—F8—OFF—F9———
                              MICRO I/O MAP

                                 Single I/O

 PLC         :  MICRO 512/00            Holdup Time  :  N/A
 Used Inputs :  040 of 256 Points       Used Outputs :  016 of 256 Points
 Next Input  :  10001                   Next Output  :  00001

Location      Type        Reference Numbers       Data     Description
                         Inputs       Outputs     Type

DISC   I/O:  MIC128   10001-10016   00001-00016   BIN   16@24V  I/O 12 RY
INT/CTR IN:  MIC140   10081-10088         -       BIN   8 INTPT/CNTR INP
TMR/CTR IN:  MIC147   30001-30001         -       BIN   16BIT TMR/CNTR VAL
ANALOG I/O:  NotAvl         -             -
DATA TRANS:  NotAvl         -             -
```

Figure 129 Entry Level Single Mode I/O Map

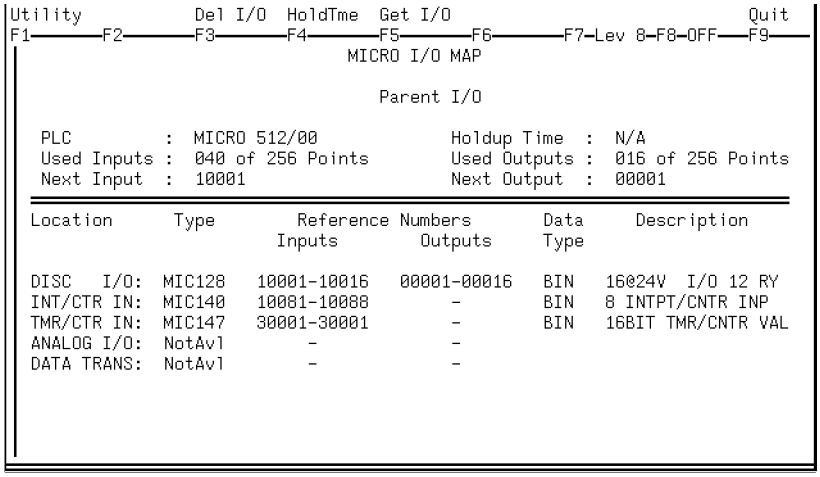


Figure 130 Initial I/O Map Parent Selected

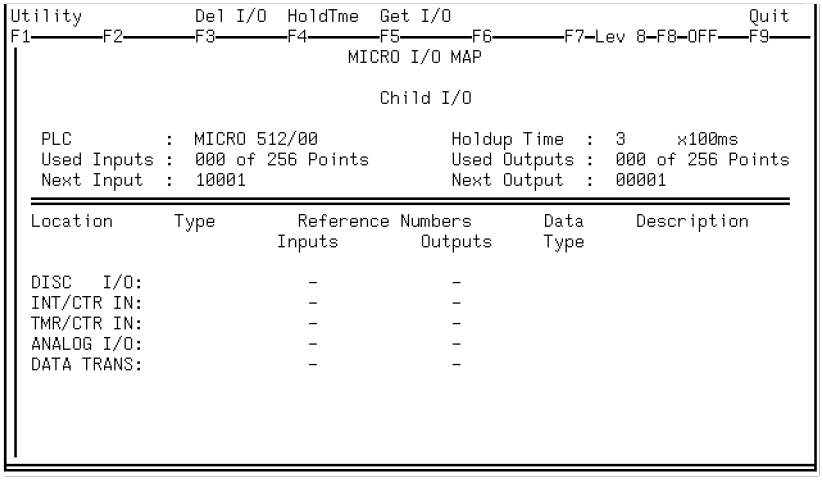


Figure 131 Initial I/O Map Child Selected

To display the list of available I/O types type < **?** > and observe the display.

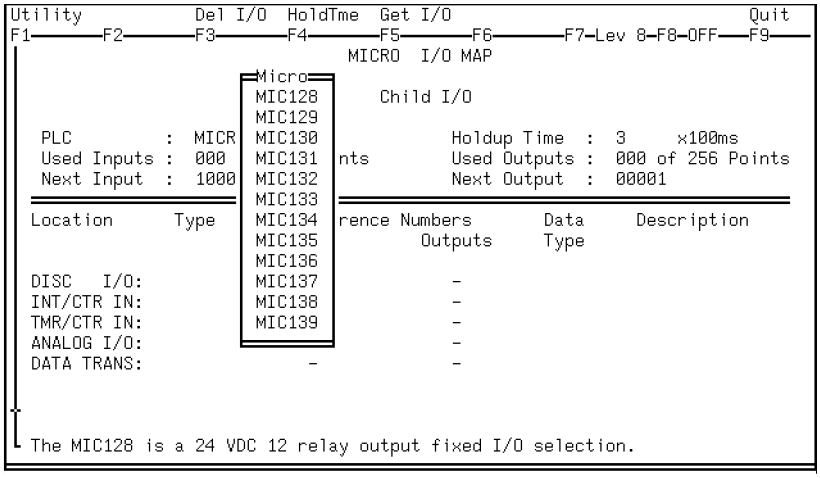


Figure 132 Discrete I/O Module Selection List

Available here as in the Compact I/O Map you can display the General I/O wiring groups by pressing **<Alt H>** with the cursor on the module of interest. The next figure illustrates the MIC128 selected.

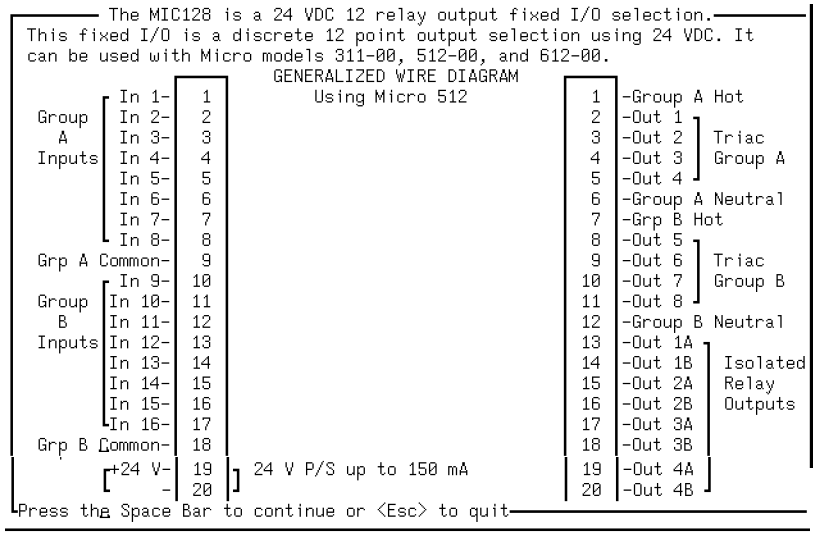


Figure 133 MIC 128 I/O Wire Diagram

You have to complete the configuration by defining the Input and Output reference assignments you require. While filling in selected I/O you can increment sequential references by

pressing the < **+** > key i.e., the reference number assigned comes from the I/O Map Next Input or Next Output field. Typing in an input or output reference automatically resets the next available reference to the typed-in value, if it is a legal value.

Both Interrupt and Timer/Counter have only one module type available and they are posted directly by pressing the "?" key.

If there are children configured, you can display the child drop I/O by selecting the GetI/O menu entry which provides both forward and backward display of data i.e., child 2, 3, 4 and 4, 3, 2.

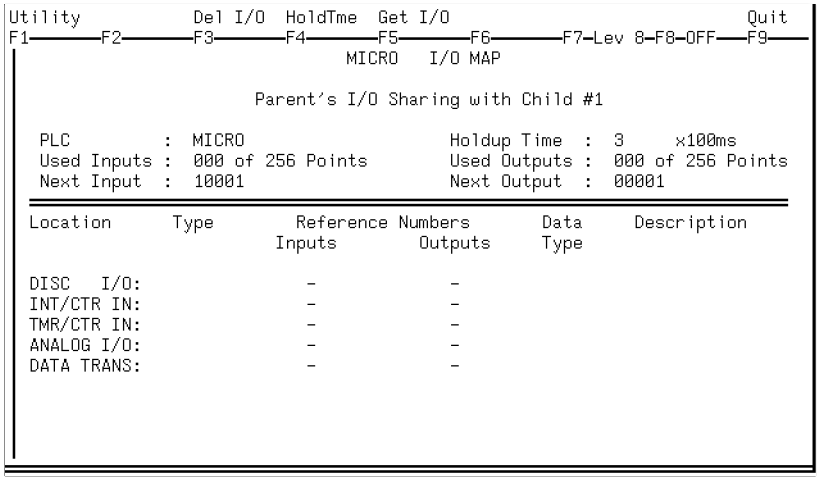


Figure 134 Child I/O from the parent Root

If you have I/O configured for a 512 or 612 PLC and are a Parent type, you can press the < **PgDn** > key to display the A120 I/O screen for Rack 2. As with the Child, GetI/O function, successive PgDn/PgUp key pressing traverses up to 3 racks.

HoldTme Lets you assign a time value to the child signal state hold up (Child I/O Only).

This value determines how long the signal state is kept at the inputs and outputs in case of a communication failure. Any number between 3 and 65535 can be chosen. This number is multiplied by the base-time of 100 miliseconds.

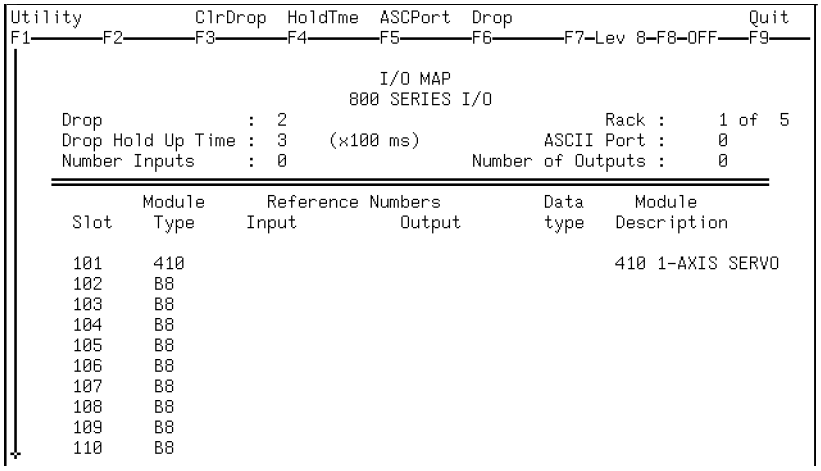


Figure 135 A120 I/O Screen

You can toggle back to the Micro I/O Map by pressing the <**PgUp** > key. The menu at the Micro level provides the standard Utility functions as well as the specific functions:

DelI/O Allows deletion of a child or children entries

GetI/O A Pulldown menu is displayed with selections for Getting Previous or Next Child if the PLC is selected in Parent Mode I/O. The shorthand keys are < **Alt P** > and < **Alt N** > respectively.

Expert If you choose the Modicon **Micro** as a controller type you will find differences in architecture, operation and terminology. To help you complete the I/O map, you may want to reference the information in ***Appendix G.***

5.6 Micro Controller Ports

"Ports"is a submenu entry of the Configuration menu. The screen is used to set the modes of communication for the RS-232 ports.

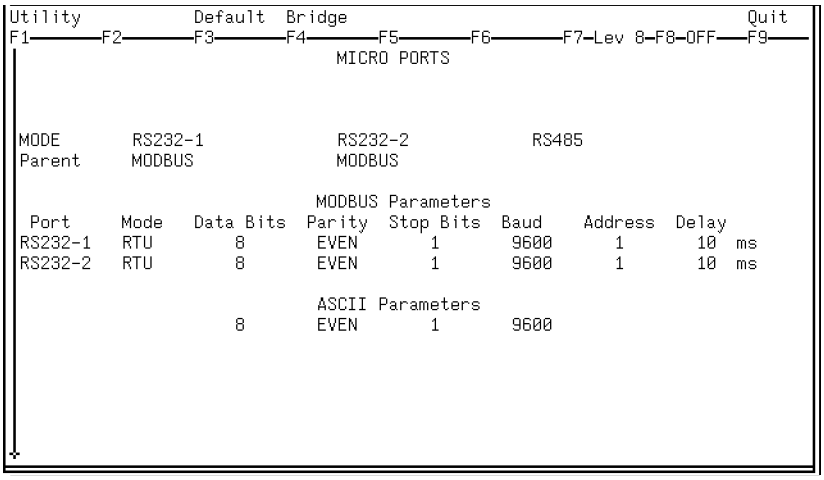


Figure 136 Modbus Port Selection Default Screen

The menu default value edit selection provides the function that can toggle the ports between Modbus and ASCII by choosing the combinations from the selection table.

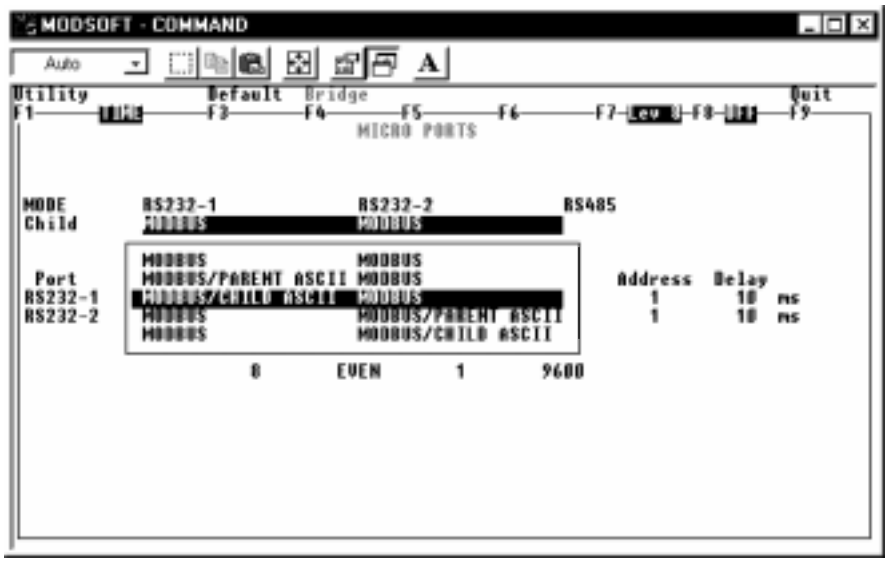


Figure 137 Ports Default and option Selection

5.6.1 Port RS232-1

Modbus port #1 may be used as either Modbus or ASCII. There are only two settings available for this port.

1) 9600, RTU, Even Parity, 1 Stop Bit, Address 1
2) 2400, ASCII, Even Parity, 1 Stop Bit, Address 1

Port RS232-2 is not available to the 311 family controller. The port may be used as either a Modbus or ASCII port. The configurable settings are:

Parity	None/Odd/Even
Mode	8-bit RTU/ 8 bit ASCII
Stop Bits	1/2
Device Address	1-247
Baud Rate	1200/2400/4800/9600

This port defaults to 9600, RTU, Even Parity, 1 Stop Bit, Address 1

☞ **Note** These settings are illegal:

7 Bit ASCII with 1 stop bit and No Parity

8 Bit RTU with 2 Stop bits and *EVEN or ODD* Parity

There is also a **RS-485 port** available for a Micro Controller configured in ***Single*** mode. It is an I/O expansion port with fixed parameters (ASCII) that can not be edited.

No more than one port per PLC can be used in the ASCII mode. If an RS-232 port is set to operate in Modbus/ASCII the RS-485 port is not used. The RS485 Port is intended to provide a communication link from Parent to Child which thus provides you with an I/O expansion. For further information refer to the appropriate Micro hardware manual.

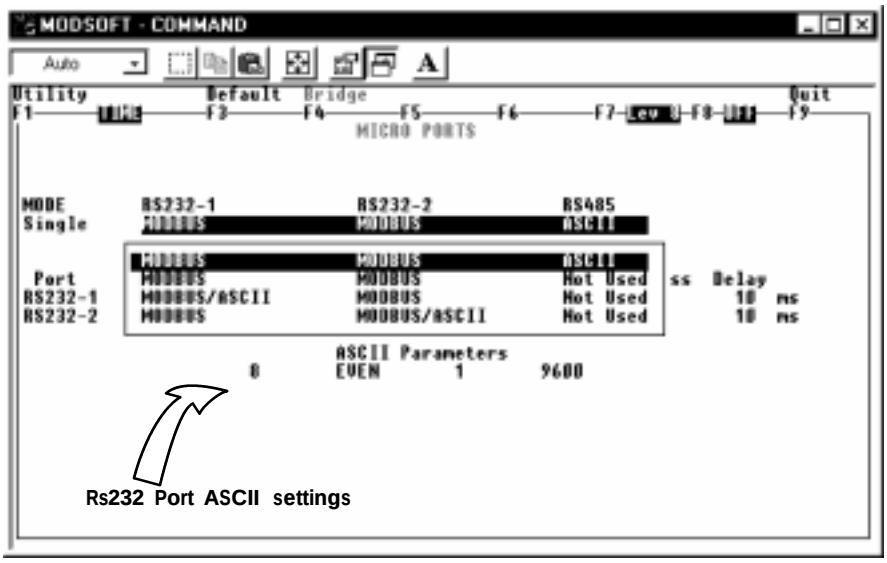


Figure 138 RS485 Port Availability

5.7 Quantum Controller I/O Map

When Quantum is the configured PLC Type, the Configuration I/O Type parameters are displayed as:

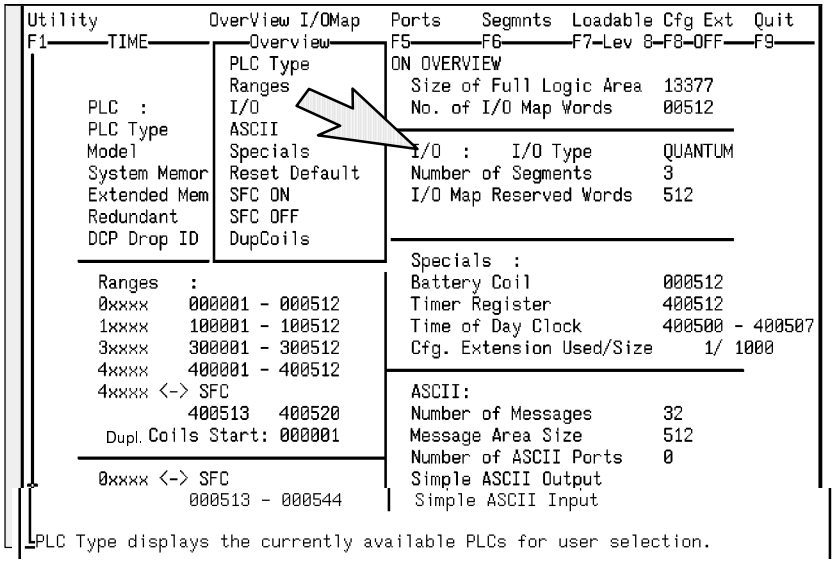


Figure 139 I/O Section of Configuration with Quantum Selected

5.7.1 I/O Map Display

The configuration Menu `I/O Map` displays a Quantum I/O specific Map.

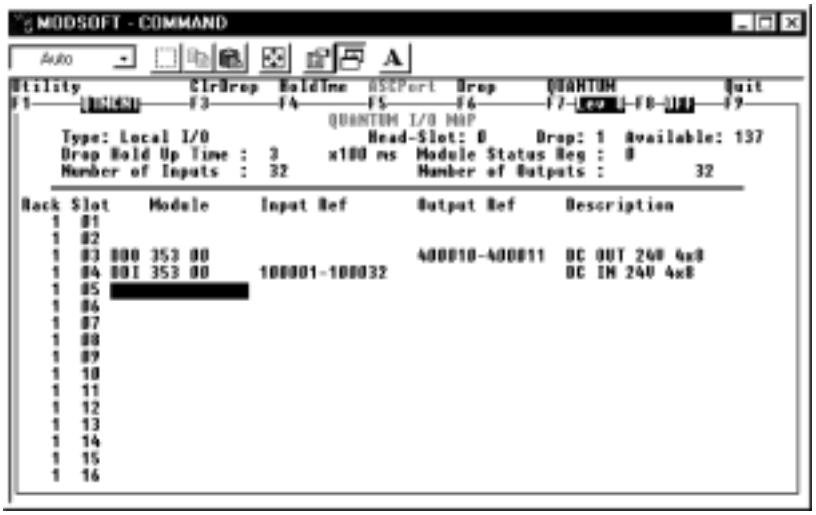


Figure 140 Quantum I/O Map with example Module Information

When the Module Type is selected the module reference numbers to be mapped are entered in 6 digit input/output fields. The initial display is for drop 1, Head-Slot 0, Local I/O.

5.7.1.1 Quantum I/O Map Head-Slot Number

The Head-Slot number is assigned according to the housing slot in the local drop (drop 1) of the device to which Remote I/O or Distributed I/O drops are connected.

Remote I/O

The Remote I/O Head end device can be a 140 CRP 931 00 or 140 CRP 932 00.

The Head-Slot number for the local drop (Drop 1) is always 0.

The Head-Slot number for Remote drops (Drops 2-32) is the slot number where the CRP Module has been placed in the local drop. All remote drops on that line have the same head number.

Distributed I/O

For Distributed I/O, the head end can be the Modbus Plus port in the Controller or either of two NOM modules (140 NOM 211 00 or 140 NOM 212 00), for a total of three possible networks.

1) Controller Port = Slot Number of the Controller

2) Slot Number of NOM # 1
3) Slot Number of NOM # 2

5.7.2 Quantum I/O Module Select

For a list of Quantum Series I/O Modules, place the display cursor on the Module field and press the **?** key. As you move the cursor over each Type the bottom line display summarizes the I/O Specification.

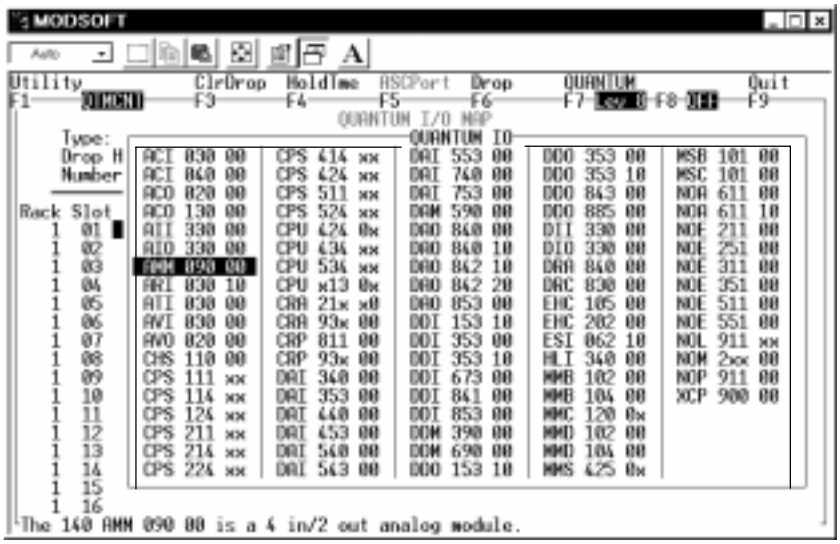


Figure 141 Modsoft Quantum Series Module Select List

☞ **Note** Although selectable, Intelligent modules must not be placed in a level 2 local rack.

When the number of modules exceeds the viewing window size, you will have to use the cursor left and right keys to bring the hidden part into view. This condition is made apparent by the << >> brackets in the top border of the list.

After selecting the module it is posted to the I/O Map display at which time another level of help is provided. Place the cursor on the selected module in the I/O map and press the <**Alt + H**> keys and a Pin-Out and Module Specification summary is displayed. To display this additional help information use the PgUp and PgDn keys. An example of the information

available for the Quantum DDO 353 module is provided for your observation. You can also type in the module parameter as desired.

☞ **Note** A fully constructed I/O Map must not exceed a size of 19043 entries.

Sheet 1

```
The 140 DDO 353 00 is a 24 VDC 32 point output module.

                    Output 2 -  2    1  - Output 1
                    Output 4 -  4    3  - Output 3
                    Output 6 -  6    5  - Output 5
                    Output 8 -  8    7  - Output 7
                  Pwr Group A - 10    9  - Group A Return
                   Output 10 - 12   11  - Output 9
                   Output 12 - 14   13  - Output 11
                   Output 14 - 16   15  - Output 13
                   Output 16 - 18   17  - Output 15
                  Pwr Group B - 20   19  - Group B Return
                   Output 18 - 22   21  - Output 17
                   Output 20 - 24   23  - Output 19
                   Output 22 - 26   25  - Output 21
                   Output 24 - 28   27  - Output 23
                  Pwr Group C - 30   29  - Group C Return
                   Output 26 - 32   31  - Output 25
                   Output 28 - 34   33  - Output 27
                   Output 30 - 36   35  - Output 29
                   Output 32 - 38   37  - Output 31
                  Pwr Group D - 40   39  - Group D Return

Navigate using <PgDn>/<PgUp> or <Esc> to quit
```

Sheet 2

```
The 140 DDO 353 00 is a 24 VDC 32 point output module.

Specifications:

Module Topology:
   Number of Outputs             32
   Number of Groups              4
   Points/group                  8
   Working Voltage Range         19.2-30VDC
   Output Current Drive          .5A/output maximum
   Indicator                     1 active status LED, green lite
                                 when valid data transfer
                                 32 output status LEDs
                                 1 fault LED, red when group fuse or
                                 no field power.
Power Supplies:
      Bus Power
        Typical                 250mA max. from I/O bus
        Maximum                 650mA max. from I/O bus
      External source           .5A maximum per output point

Navigate using <PgDn>/<PgUp> or <Esc> to quit
```

Figure 142 Example of Quantum Series Module Information Via <Alt + H>

5.7.2.1 Quantum Drop. Add and Delete

A `Delete Drop` item is available in the `Drop` menu of Quantum I/O Map display. Modsoft requires that you confirm the delete, thus preventing inadvertent loss of work. *It is not possible to delete drops from non-Quantum I/O Maps*

Figure 143 Configuration I/O Map Drop Function with Quantum

In Modsoft, a cleared drop (`ClrDrop`) is deleted when the user exits from the configurator. The exit process also removes the drop from the segment scheduler. It is necessary when a drop is deleted to delete any and all references to it in the segment scheduler file.

The user will not be allowed to delete drop 1 of head 0, (the local drop). Special treatment is afforded the VME 486 controller WHERE THE FIRST DROP IS NOT LOCAL.
Drop editing is not allowed in on-line mode and the ClrDrop function is disabled.

5.7.2.2 Drop Type

If you are not in a Local Drop, you are presented with a pop up selection list that allows you to pick from two specific remote I/O functionalities.

Quantum or 800 I/O When 800 type is selected the I/O map displayed is for the 800 series I/O. The configuration presumed by this I/O mapping is that the hardware is driven by a Remote I/O head similar to the S908 and an 800 series I/O Map is displayed. If `QUANTUM` is selected the Quantum I/O Map is displayed.

The Remote can further be assigned to function ***type*** as
Quantum I/O or

800 I/O or
SY/MAX S908 I/O

When you complete this assignment the I/O map that represents the selections you have assigned is displayed for your Module and register assignments.

If you choose Distributed, you must further assign the properties of:

Read & Write Drop
Read Only Drop

5.7.3 I/O Map Quantum Specific Pulldown

When you configure a Quantum controller the I/O Map provides the same editorial selections as for other controllers and, in addition, has a Quantum Series specific pulldown menu:

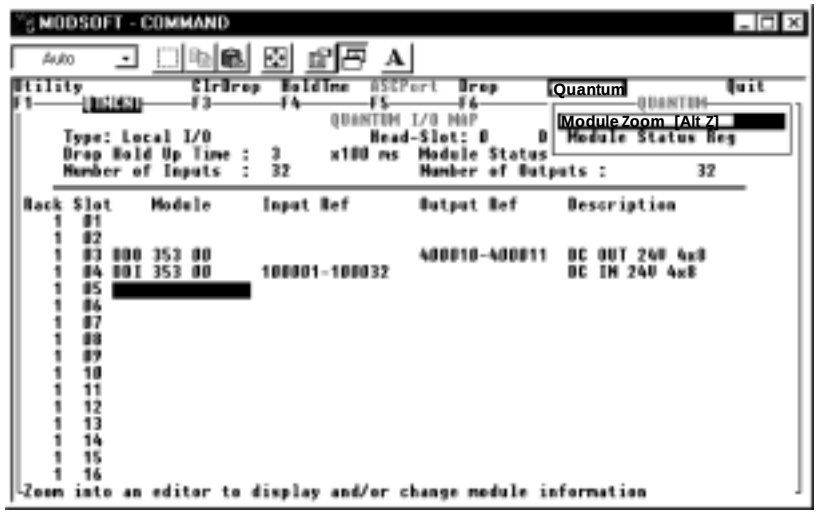


Figure 144 Quantum Specific Function Menu Pulldown

5.7.3.1 Module Zoom

This menu entry allows you to enter the **I/O Map module editor**. For example, the DC-Output module illustrated has a map editor with a single page display. Put the screen cursor on the desired module then select the *Module Zoom* function. The summary information on the top line ie.Head - Slot, Drop, Slot are not editable and the cursor is initially placed in the *Output Type* field.

The editor menu functions `Hex, Dec, Bin` and `Goto`, allows you to change the user defined value expression in engineering units and access multiple pages when they are present.

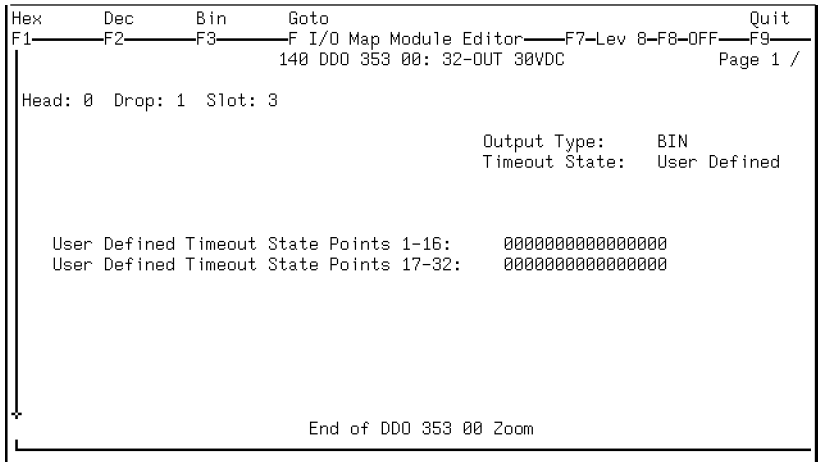


Figure 145 Example of I/O Module Zoom

The *Slot* field is filled in automatically by the software and indicates the slot in which the module resides for this screen.

The *Output Format* field allows the choice of mapping outputs from Ladder Logic to either 984-800 series I/O modules or Quantum I/O modules.

Warning You must properly include this field in your configuration and download it to maintain the proper relationship between output points for the two different I/O types.

The Output Type field allows the selections of BIN (binary) or BCD (Binary Coded Decimal).

The *Timeout State* field gives you the option to select from the list:

Last Value
User Default

Timeout Parameters in Hexadecimal notation are entered according to the module requirements. If you try to enter a value while online a system message will inform you that you can not change the configuration while online.

5.7.3.2 Module Zoom for more complex Modules

Depending on the function the module is to perform, the data you may supply can be expressed on one page as with the DDO 353 module or may require multi page entries as with the 140 ARI 030 which has a five page parameter entry zoom screen.

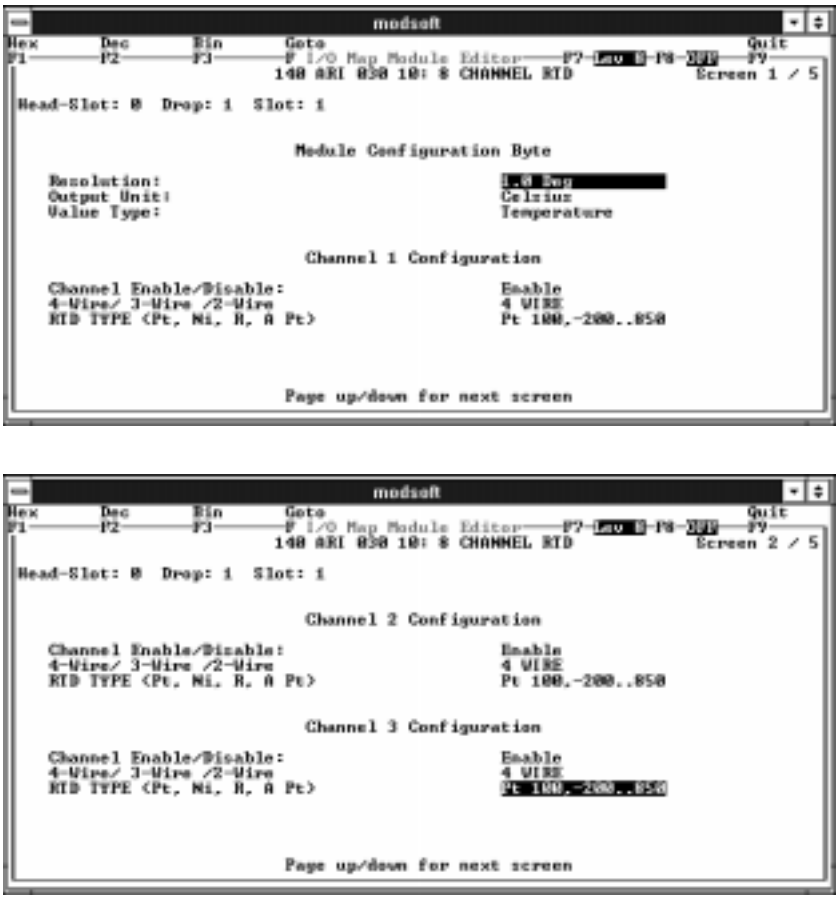


Figure 146 Example of Multi Page Module Zoom

All fields offer selection lists for the possible settings. In the example illustrated above, the cursor is positioned in the first variable field on page one. You may choose the default settings or press the **Return** key to display the selection list.

You can view and edit successive pages by pressing the **PgDn** (or PgUp for preceding pages) key. In this example, on page two, there are four fields for each analog channel. The remaining pages are duplicates of page two that account for the total of 8 data channels. The exact settings depend on module type and application.

5.7.3.3 Module Status Register

This menu entry allows you to assign the 3x register that defines the start of a table of 3x registers in which I/O mapped Module status is available. You may either enter the 3xxxxx value or the value 0 (indicating no choice). The value entered is displayed in the summary information on the top of the Quantum Series I/O Map. Each module in a rack reports status and fault information in an 8 bit byte, therefore one word of table conveys two module status words.

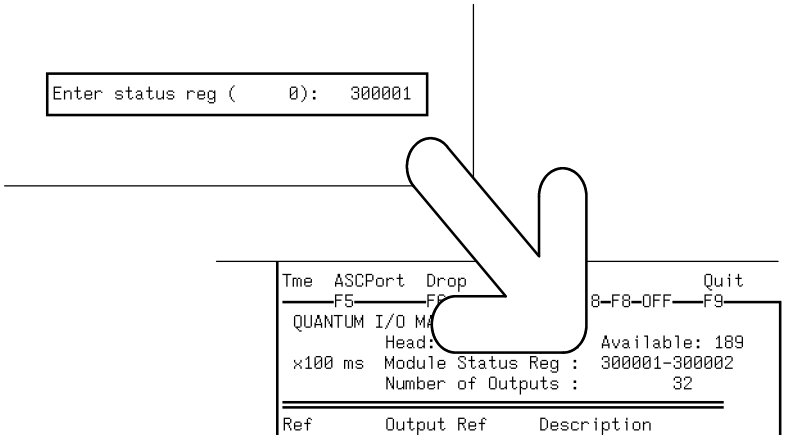
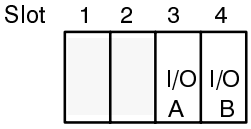
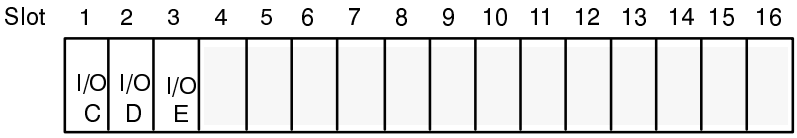
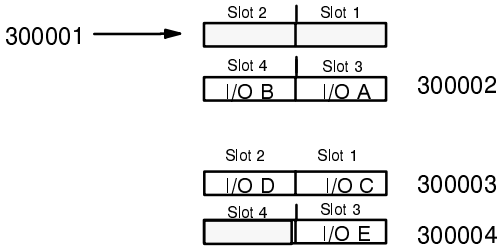


Figure 147 Status Register Input Field and Configuration Update

If you choose to display or develop a program using these values the table/module relationship is given in the following example:







Given the above example configuration, if you select 300001 as the starting address of the status table and there are no I/O modules in the first 2 locations, the first I/O module status is found in the least significant byte of the second word i.e position 3. The table fills until the last I/O Mapped module with status is found.

☞ **Note** The bit pattern reported in each status/error byte is dependent on the module type.

5.8 Momentum - M1 Controller

5.8.1 Configuring the Type

Select `PLC Type` from the `Overview` Menu to display the PLC selection list. After selecting Momentum you use the Model selection to choose between:

M1
Magnum

These are the memory selections for you to choose between:

M1	Magnum
2.4k	10.0k
12.0k	18.0k
18.0K	

After entering one of the Memory sizes the cursor is positioned on the main menu select line, and the data you have entered is displayed in the `PLC` section of the Overview screen.

5.8.2 Configuring the I/O Map

I/O Map for a Momentum M1 controller requires two different mappings:

I/O Base Module This I/O Map describes a base Momentum I/O module that you select from a list of possible choices. You are limited to only one choice (4 different types DC I/O AC I/O Analog I/O and Specialty I/O.). In relation to other Modicon product terminology, this is the equivalent of a “local” drop.

I/O BUS This I/OBUS Map is enabled from the Base map by selecting the Add Drop (or subsequently Next drop) function . This map describes I/O modules (up to 128 modules) that are attached to the base module via communication cable and may be thought of as “remote”.

Tip Only 80 modules (nodes) are allowed in Modsoft 2.5

The IOBUS can have a maximum of 2048 bits of I/O in the interbus-s bit stream. Thus the total length of all I/O nodes in the IOBus network must total to less than or equal to 2048 bits. This is a departure from the usual sense of used I/O points in the IO map.

5.8.3 Base Momentum I/O Map

This configuration function provides the interface mapping between the input and output control points and the program that controls them. The initial screen is labeled “MOMENTUM”, and acts as a template. This template has previously been used for Quantum and when used for Momentum the notion of rack and slot are no longer valid because the hardware configuration of the base module, Control Processor and communications card are physically interconnected.

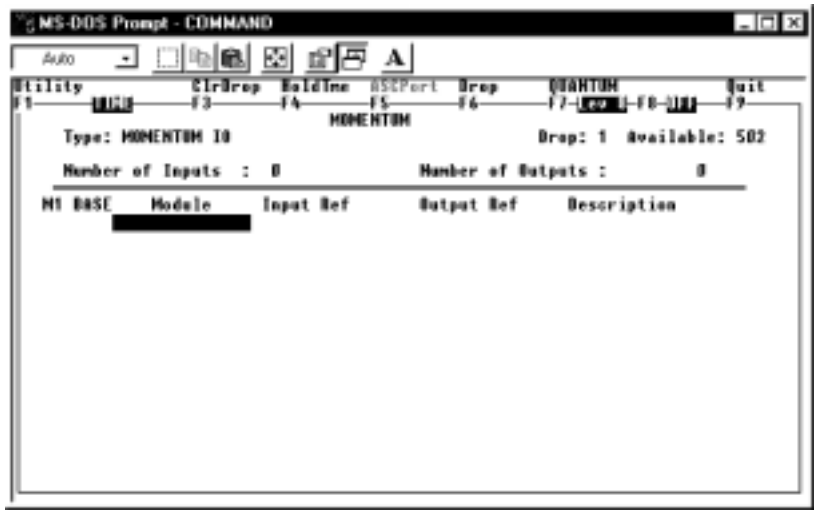


Figure 148 Initial Momentum I/O Map Screen

Only one module can be entered. An error is displayed if you try to go beyond one entry.

As with previous I/O Maps, if you position the cursor in the Module field and press the ? key you enable the display of the Momentum module select list. Additionally, you can directly enter data in the Module field if you know the number without viewing the list.

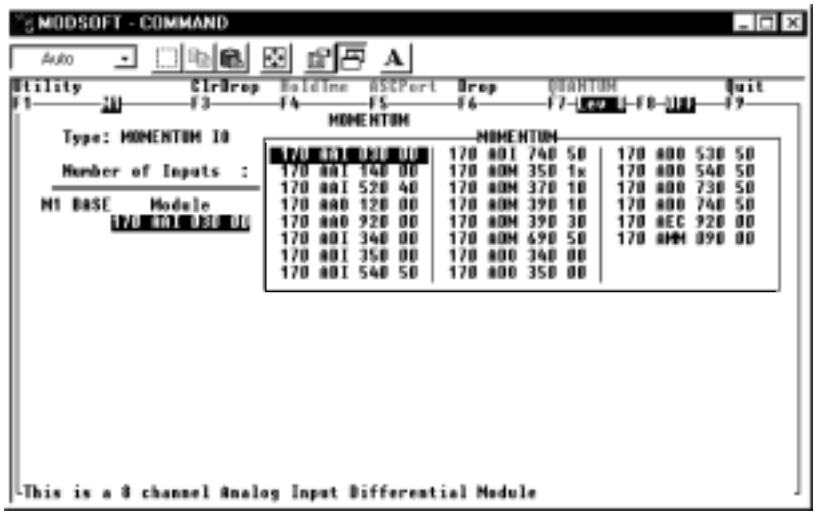


Figure 149 Momentum Module List in Modsoft 2.6

Also available in the Momentum I/O Map are the Pin-Out and Specification for the selected module. to view this information just press “Alt h” with the cursor on the Module field. These help screens may be on multiple pages which you view using the `PgUp` or `PgDn` keys.

The initial template indicates the Module `Type:` as `Momentum I/O` and is essentially the local I/O for the base module you selected.

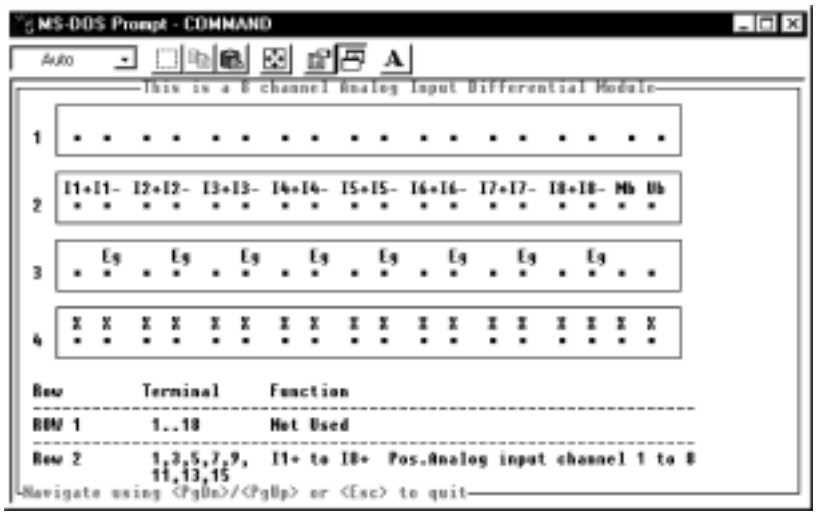


Figure 150 Example Of Pinout Help For AAI 030 00

5.8.4 Setting Module Parameters

Momentum I/O modules require certain enabling and operational parameters. With the cursor on the Module of choice, press the `Alt` and `Z` keys to invoke the zoom screen for the particular module. The zoom screen acts as a calculator or prompt for you but does not place values in the PLC. You must complete the action of entering values in the proper registers.

One example of a zoom screen is represented by the following 2 screen Zoom

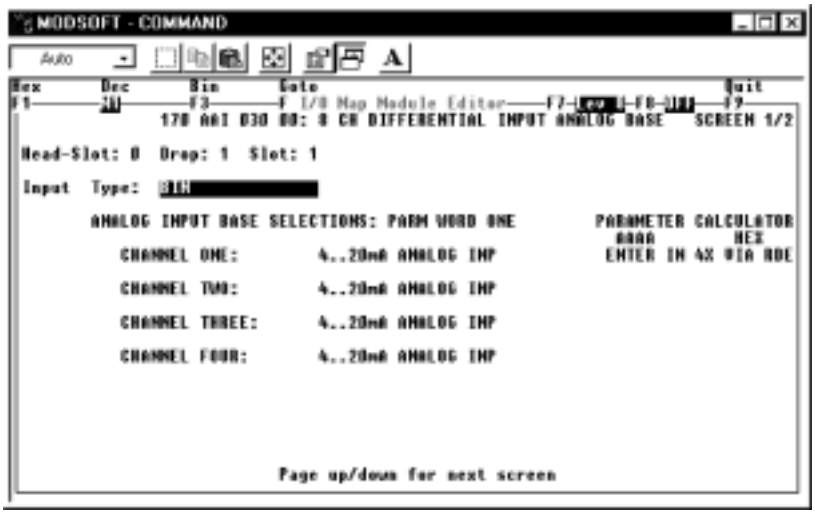


Figure 151 170 AAI 030 00 Screen 1

You have the option of selecting between BIN (binary) and BCD (binary Coded Decimal) by pressing the return key with the cursor on the Input Type field. Additionally with the cursor on a Channel field you may select form the 4 options:

Disable Channel x
4..20mA Analog Input
+/- 20 VDC Analog Input
+/- 10 VDC Analog Input

Additional screens are indicated in the upper right corner of the display. In this example there are 2, and the second one appears as follows.

Figure 152 170 AAI 030 00 Screen 2

This Zoom also allows you to select the data display format from main menu selections of Hex, Dec or Binary and you are offered the ability to “GoTo” a specific screen on a multiple screen function.

5.8.5 IOBUS

This network is comprised of a number of I/O devices, interconnected through communications cables and communications ports on the hardware devices. You may configure the I/O BUS after declaring the PLC Type and it’s Base I/O map to which you either “ADD a Drop” (first instance) or select “Next Drop” (in established configurations).

Modsoft supports :

2048 points for the 256k PLC

4096 points for the 512k PLC

When selected, a default template similar to The I/O Base Map is displayed.

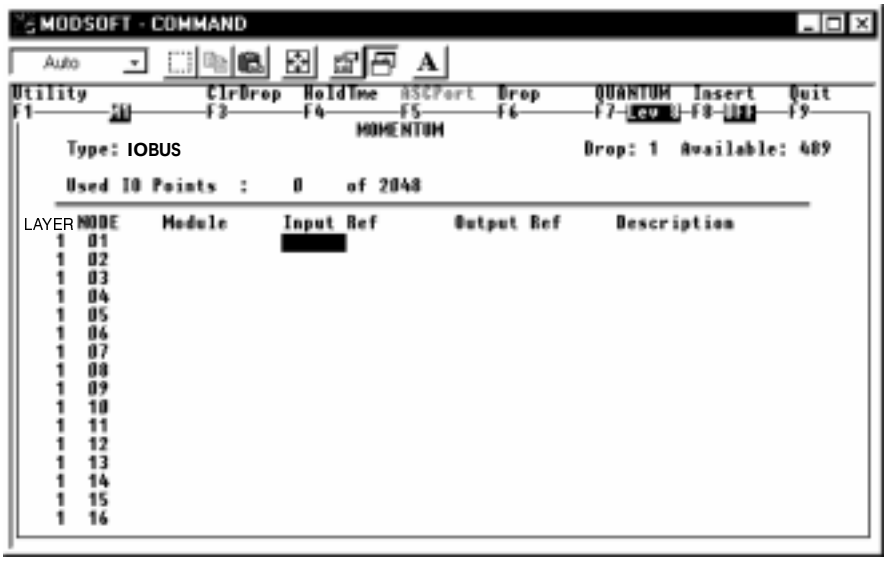


Figure 153 Initial IOBUS MAP Default TemplateDisplay

5.8.6 I/O Bus Module Selection List

With the display cursor on the Module field if you press the ? key, all available I/O modules are displayed in a table. This list may be a larger list than can be put on the screen at one time, in which case, the list can be scrolled to the right and left to show more columns. Modules in the list appear as given in an internal file description. This file also contains all of the known selectable default modules. This feature helps a user who has forgotten the real name of a module, but knows its I/O characteristics as provided in the description field.

☞ **Note** The above I/O Bus screen header has some variations from the usual Quantum I/O Map.

1) The header does not represent Input and Output counts in two separate fields. The value displayed is the number of used I/O Points in the Interbus-s style module network. The number is not the simple sum of the Inputs and Outputs. It is the sum of the individual modules I/O width. The Interbus-s module width is derived from the larger of it's Input or Output points.

2) The Left 2 Module definition columns are the Layer and Node number.
You can assign from 1 to 16 Nodes per Page.

3) Modules may not be placed in Node positions that would create an empty Node position.

5.8.6.1 Page Scrolling

You can press <PgDn > or <PgUp > to get to additional pages of IO/MAP for this IOBUS. The scrolling will not continue into empty pages. If You press <ctrl + Page Down> the Last page in the list is opened. Likewise pressing <ctrl+Page Up> returns to the First page in the list.

One notable departure from other Modicon types is the `Layer` and `NODE` data on the left of the screen. The number of modules supported per M1 type is:

For the M1 up to 128 modules or up to 2048 I/O points (whichever comes first) are displayed in 8 screens of 16 modules

For Example a module map that requires 2 pages follows:

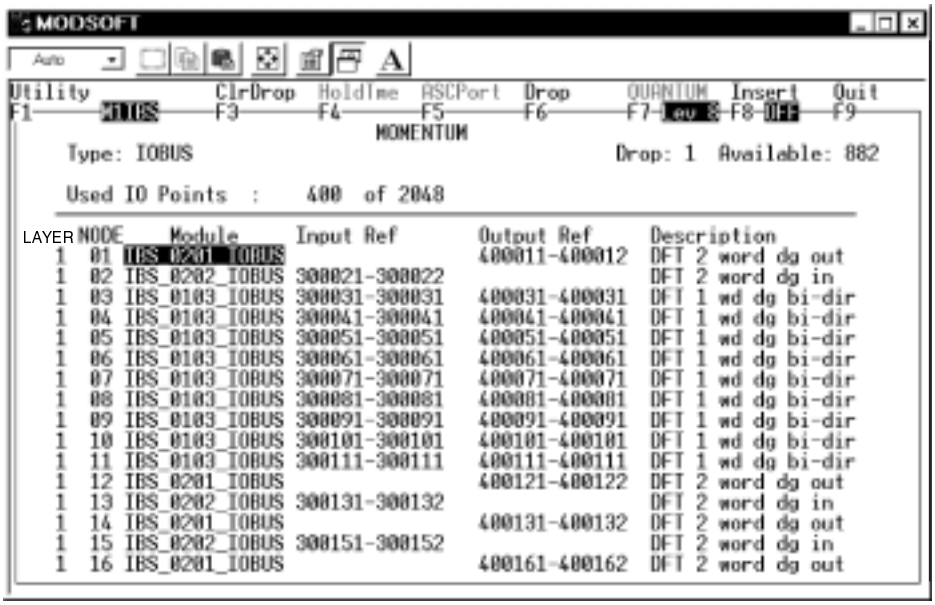


Figure 154 I/OBus Page 1

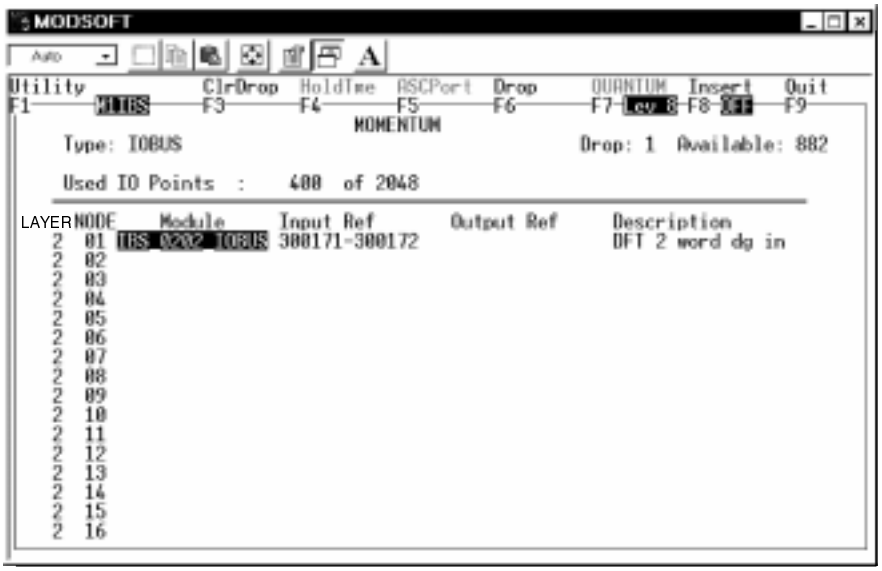


Figure 155 I/OBus Page 2

Also available in the Momentum I/O Map are the Pin-Out and Specification for the selected module. To view this information just press “Alt h” with the cursor on the Module field. These help screens may be on multiple pages which you view using the `PgUp` or `PgDn` keys.

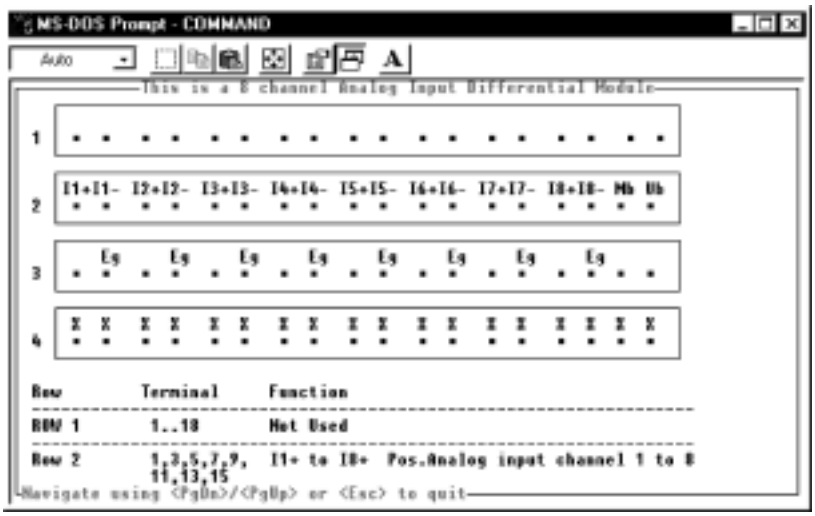


Figure 156 Example Of Pinout Help For Module AAI 030 00

5.8.7 Read Drop Personality

There is a special read Personality operation for the M1 in which the PLC (when Stopped) scans the network of I/O and creates a table that corresponds to the installed modules by type and location. Modsoft must read this PLC based data and use it to compare the existing configuration I/OMAP for differences. When the PLC is stopped, a * in front of the module type in the I/O Map display indicates a mismatch in the personality.

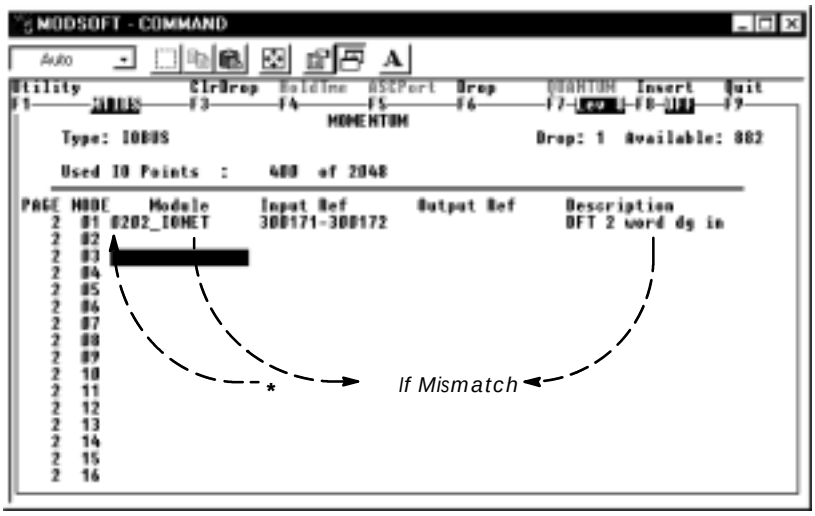


Figure 157 Module Vs. Description check

5.8.7.1 Insert and Delete In I/OBUS

By definition, there is never an empty slots in the I/O Bus structure. If a module is deleted, <Del key> all modules in the IOBUS Map after the deleted entry are shifted up in position. If a module is inserted (use the new `Insert` Function key item in menu), all of the modules are shifted down in the drop to make room for the new module to be inserted. Invoking the `insert` creates a blank space which you must fill with a module from the list. The list is automatically opened for the selection. If you neglect to select a module from the list, the slot on the screen will be closed up.

This functionality is available both on-line and off-line. In the on-line mode, any changes to the IOBUS map are only available if the controller is stopped. When additions and deletions are made the PLC memory is adjusted to allow the new module in the correct position. If a module is deleted, the slot position on the screen is filled with the module below it, if one exists.

Appending the I/O Map configuration must be done at the first empty line after the last occupied line. Modsoft gives an error message, "`Empty node positions not allowed`" if

you attempt to put an IOBUS module anywhere where there would be an empty slot between modules.

5.8.8 Module ID Subtype

Modsoft uses a subtype designator for each of the two categories of modules that can be found in an IOBUS configuration.

1) Momentum I/O with Interbus-s communication mounts
2) Existing Classic TIO

In order to identify the product, the subtype is stored in the PLC along with the I/O Map information for each module. This identification allows the panel to display the data associated with the correct module including help and zoom.

5.8.8.1 Adaptable Interbus-s

The Interbus-s modules are numbered (4 digits) according to their I/O capability, a Module listed 0103_IOBUS for a module with one word **in** and one word **out**, a sub type is needed to know what kind of module it is. The sub type has no meaning to the controller. It takes the combination of both the I/O type and the subtype to find the module name and associated data in the module sys file. The following meaning is applied:

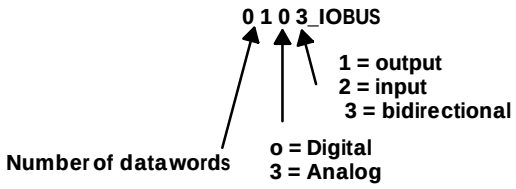


Figure 158 M1 Sub Type

5.8.8.2 Classic TI/O

The Classic TIO modules have part numbers on each label. The last four digits of the part number are used for the subtype. For example given:

P/N 042 703 807

Modsoft would store the Hexadecimal number 0x3807 as the subtype. Obviously there must be no duplication of part number data in the last 4 positions.

If a subtype is not found in the database, the module is displayed in modsoft as unknown.

5.8.8.3 Unknown IOBus Modules

If you have a module installed on the I/OBUS and the module Identification is not currently in the database, the Modsoft IOBUS map display will look like:

```
Not in file (Input Ref)   (Output Ref)  Unknown Module
```

The Not in file type has no No I/O length or Interbus-s id.

5.8.9 Module Parameter Entry

IO Bus modules can require Zoom parameters just as previously described for the Momentum I/O base. Here is an example of the Zoom (Alt z) for the 0101_IOBUS Module.

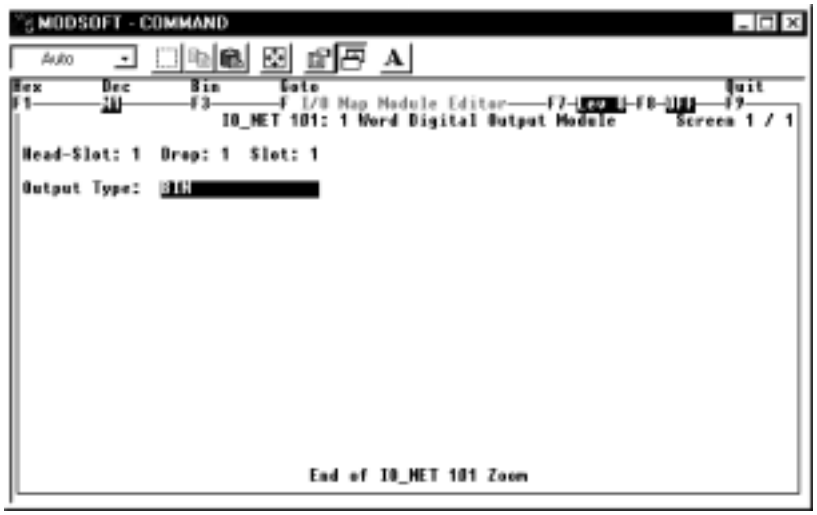
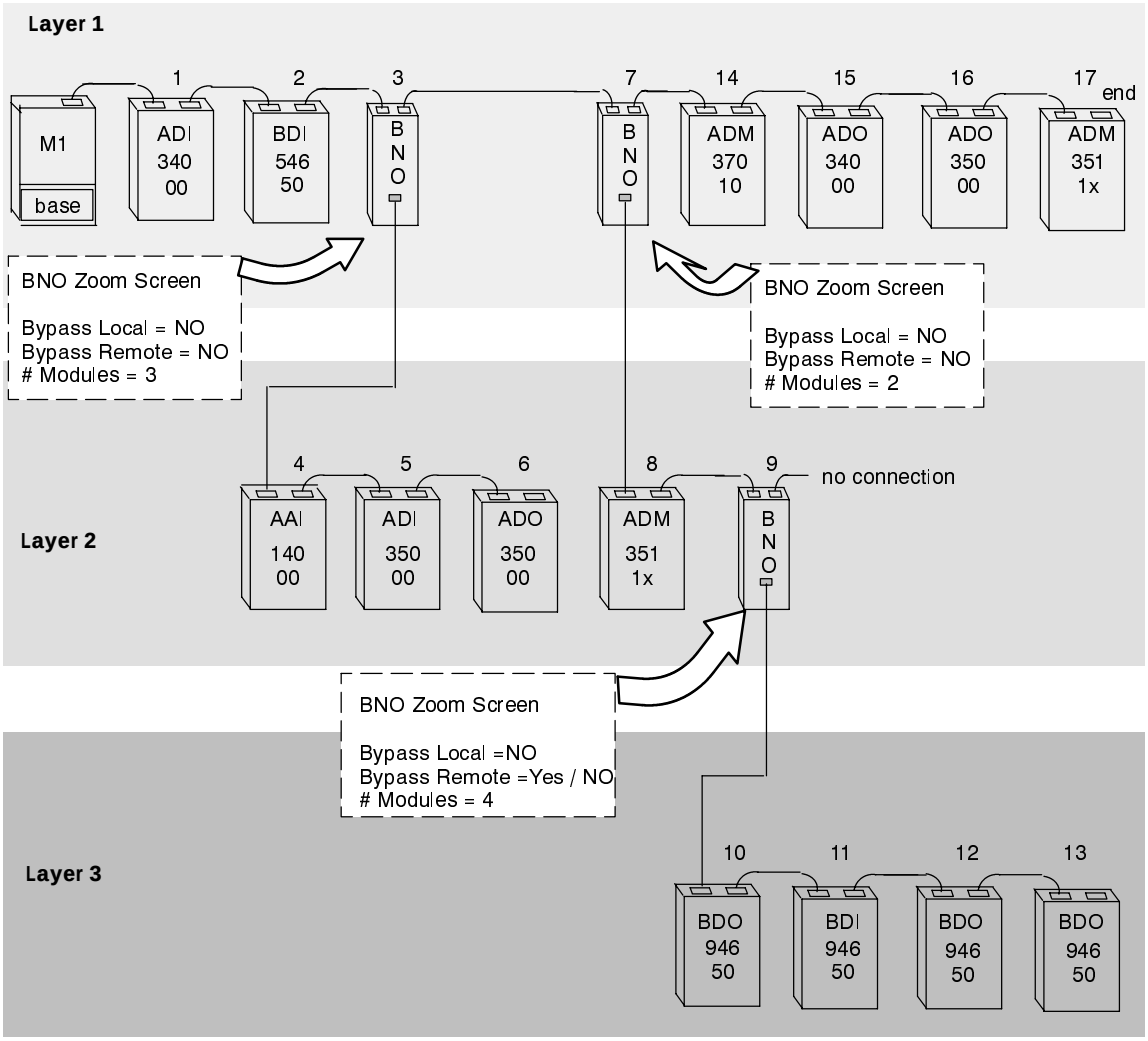


Figure 159 Example Of I/O Bus Module Zoom

5.8.10 Implementationof a BNO (Branching) Module

The Momentum BNO module provides a hardware technique that allows you to break up the serial IONET into shorter logical groups of modules each of which can be bypassed while the rest remain functional. This means that when a fault is detected in a large network, the Modsoft operator can set the network to branch around the faulty module. To properly identify a specific BNO module and it's associated I/O a **layer architecture** is implemented. The following sample drawing demonstrates the *layer* topology.



Note: Always count a BNO when placed on the Local cable of an earlier BNO module.

An example of a Modsoft Configuration and I/O Map assignment that implements a BNO IO-Bus as illustrated above would be defined as follows:

Step 1 Select Magnum/M1 as the PLC Model (With proper memory)

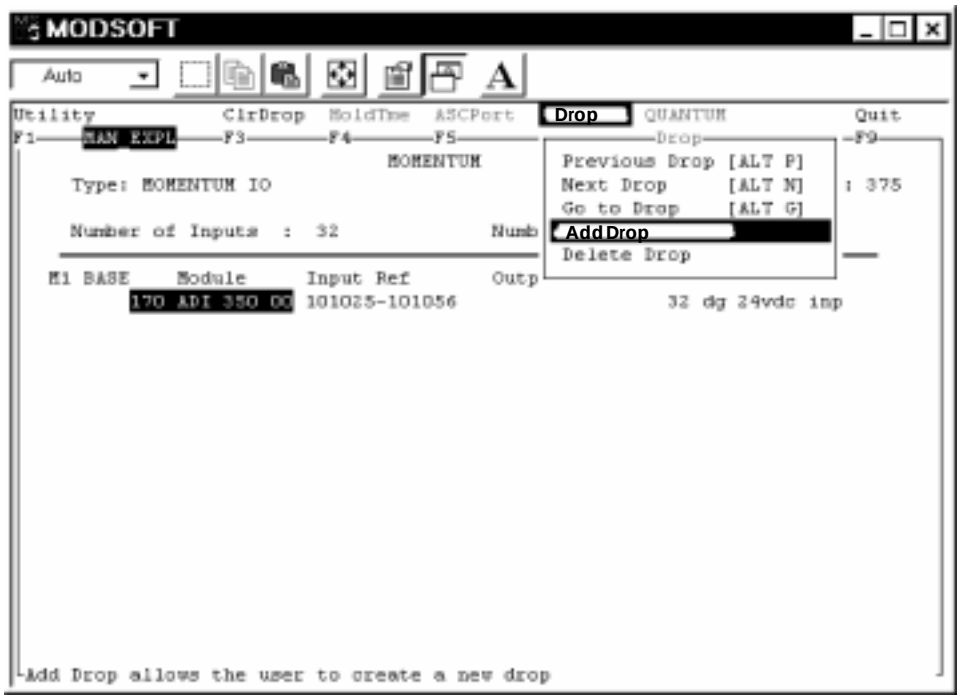
Step 2 Set Overview → I/O to Number of Segments = 2

(and allow enough memory space to cover the I/O being configured). In this example

I/O Reserved Words = 500

Step 3 I/O Map the base module

Step 4 Use Drop Menu and select Add Drop



Step 5 Insert the first 2 modules (nodes 1 & 2) ADI 340/BDI 546

Step 6 Insert the first BNO Module
Alt Z on BNO and set the zoom parameters (No, No, 3 Modules)

Step 7 Enter I/O Modules AAI 140, ADI 350, ADO 350
Note the screen indication of Layer 2 (refer to next figure)

Step 8 Insert second BNO Module
Alt Z on BNO and set the zoom parameters (No, No, 2 Modules)

☞ **Note:** If a BNO is configured Off the local branch of a BNO you must include this in the # of Modules count.

Step 9 Insert I/O Modules ADM 350, BNO 671

Step 10 Zoom on the 3rd BNO Module

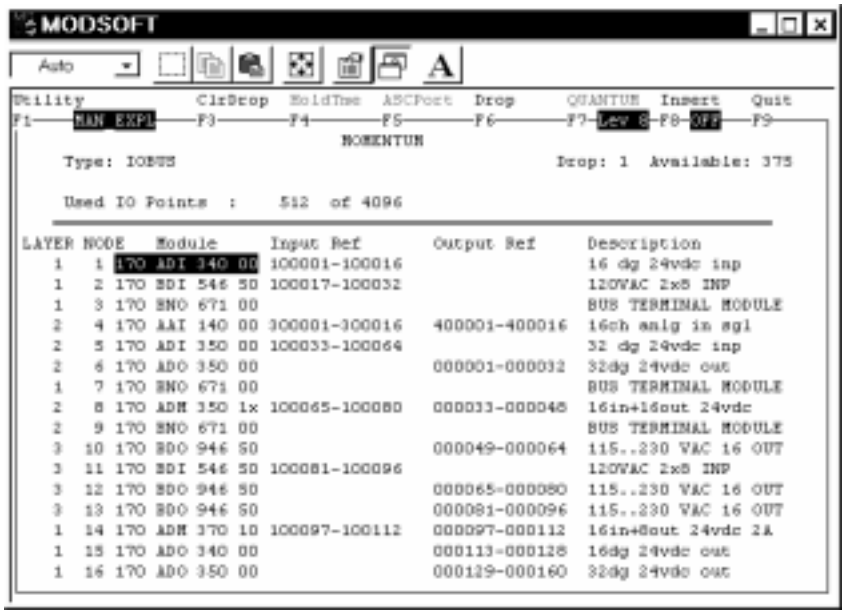
Bypass Remote = Yes or no i.e. with nothing connected the PLC internally defaults to Yes.

Bypass Local = No

Modules = 4

Step 11 Insert I/O modules BDO 946, BDI 546, BDO 946, BDO 946.

Step 12 The Layer indicator now reads 1. Insert the I/O Modules ADM 370, ADO 340, ADO 350, ADM 350.



5.8.10.1 Layer Number

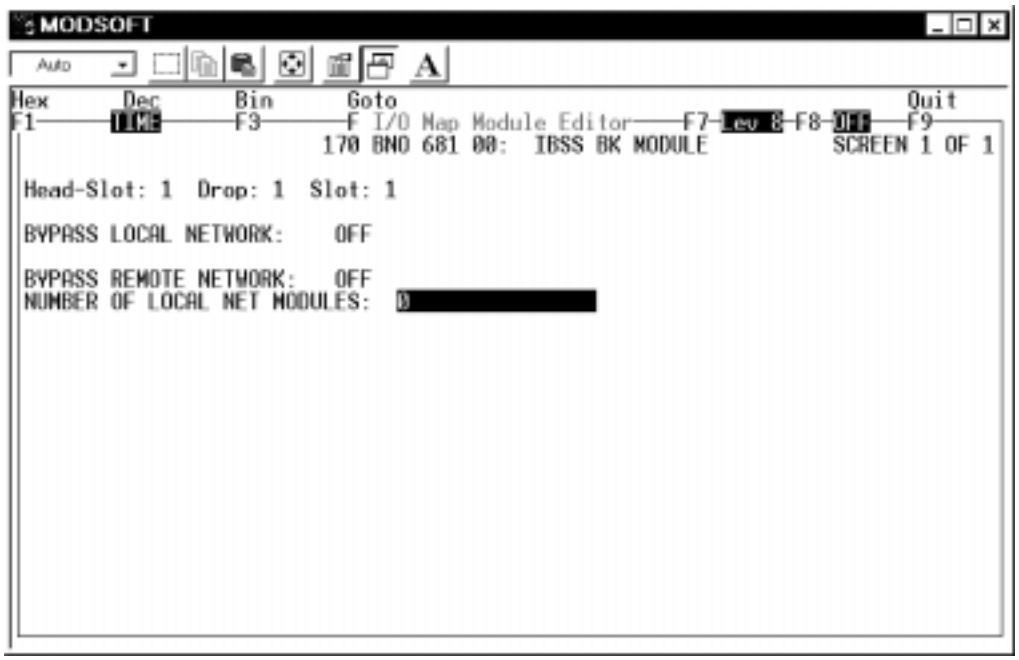
The layer number is in the leftmost column. The layer number is 1 for all modules in the top layer, including the BNO module itself. All modules after a BNO module up to the number of modules that make up the local loop for that branch are at layer two. If layer two contains a BNO module, its local loop modules are at layer three, and so on. The only limit to the layer number value is that it have a range of one to nine.

5.8.10.2 Module Numbering

The next column contains the module (node) number. Node numbers range from 1 to the maximum number of nodes, at sixteen per page.

5.8.10.3 Number of Modules on the Local bus

This value is entered by the user in the *zoomscreen(AltZ)* for the selected BNO module (see Zoom screen below). The range of this number is 0 to 255. Depending on this value (N), the N successive modules (nodes) in the I/O Map are assumed to belong to the BNO and are assigned a layer number one higher than that of the BNO module itself. When N modules layers are marked, the layer number for successive modules drops back to that of the BNO module. Inside the I/O Map itself, that is within the PLC, there is no direct indication of the layer number associated with any module.



The Zoom screen above is illustrated in a default condition. If you were setting the parameters for the example shown you would set Bypass Local ON and define the number of Modules as Four i.e. 7, 8, 9 and 10. The PLC thus drives the entire IO Network but ignores these 4 modules. For operating conditions read the paragraphs below.

5.8.10.4 Insert/Delete Module from Layer

When a module is either inserted or deleted to/from a layer below a BNO, it is necessary for you to adjust the number of modules in the Branch via the Zoom screen. Modsoft does not automatically do this.

5.8.10.5 Bypass Local

Bypass Local refers to the modules in the layer(s) below the BNO module. This setting requires module Zoom to allow setting or resetting the Bypass Local feature of the BNO modules. The Bypass Local selection is greyed out except when the cursor is on a valid BNO module and the Programming Panel is online or combined mode and the PLC is stopped.

Bypass Local Flag - When the Bypass Local is set, the entire local loop of the network that is local to the BNO module will be ignored by the PLC. Modsoft then shows all modules in this layer, and sub layers, to be unhealthy.(see IO Map modules marked with *)

5.8.10.6 Bypass Remote

Bypass Remote requires the module Zoom to allow setting or resetting the Bypass Remote flag. This flag is equivalent to Bypass Local except that it applies to the so called *remote line* of the Interbus-S network. Remote refers to the line connecting the modules in the Layer (see diagram above). In fact, all modules have a bypass for their remote connection. At this time, only the BNO modules Bypass Remote flag is implemented in its zoom screen

5.8.10.7 Fault Detection

The PLC detects network connection faults when the read personality is executed. The plc returns an array of information about the modules detected, including whether a fault was detected in the lines (local and remote) exiting from the module. The plc also indicates how many faults were found. When the plc is stopped, Modsoft attempts to inform the user with this information to help in fixing the faults. Modsoft has a system message report if a fault is detected by the PLC. The node is reported, along with whether it is local or remote line fault.

5.9 MagnumI/O

5.9.1 Configuring the I/O Map

Unlike the M1, there is no local I/O for a Magnum. I/O mapping proceeds as previously described for I/O Bus. Modsoft supports up to 8192 I/O points for the Magnum controller.

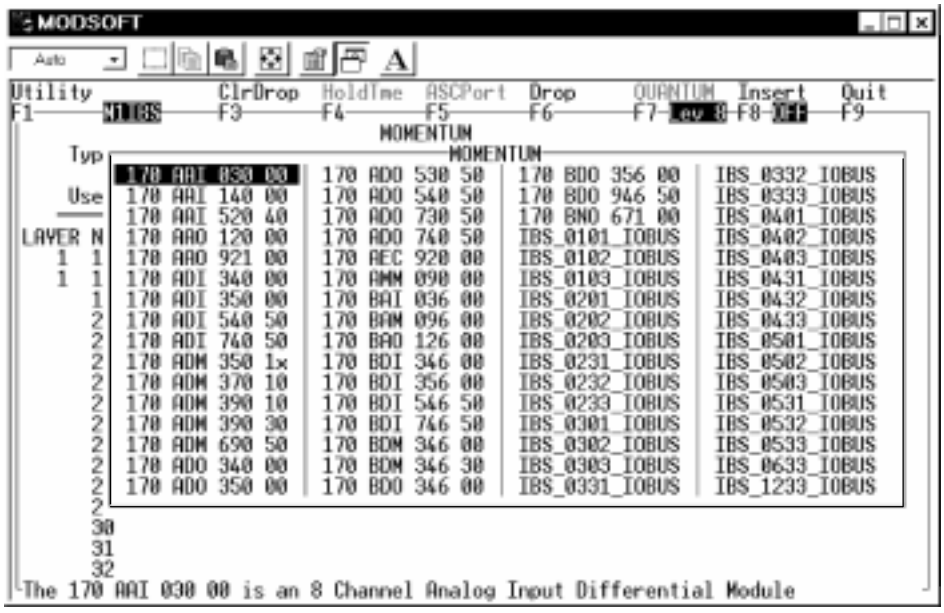


Figure 160 IOBUS Module table

You may implement up to 256 modules or 4096 I/O points (whichever is first). These will be displayed in 16 screens of sixteen modules each.

5.9.2 IR Port Support for Magnum

5.9.2.1 Port Parameter Defaults

The magnum IR port defaults to 19.2k baud. Modsoft defaults this port to the same parameters when offline. When you have selected a Magnum controller type you will see the "`Ports`" display Modbus Port 2 is set to 19200 Baud. This default value is needed because the user may create an offline project without setting the port parameters.

5.10 SY/MAX S908 Module Interface

☞ **Note** If you intend to make a SY/MAX drop, you must first use the UPF editor to set the field “Using SY/MAX S908 IO” to ***Yes.***

This interface is processed in a Quantum remote drop. Your configuration Overview I/O selection must have the number of segments set to a value greater than one.

- When you select the `I/O map` The local I/O map is displayed.
- To display a remote drop go to the Drop entry on the menu line and select `Add Drop` (first time `Next Drop` if it already exists).
- Enter the `Head-Slot` and The `Drop` Number in the on screen prompt
- You are prompted to select between Remote IO and Distributed IO.

The selection of SY/MAX is added to the selection list for *droptype*<u>*after*</u>you select Remote IO. The New selection box appears as follows:

QUANTUM IO
800 IO
SY/MAX S908 IO

The following is a partially filled in SY/MAX I/O Map. The module numbers and descriptions are per SY/MAX models.

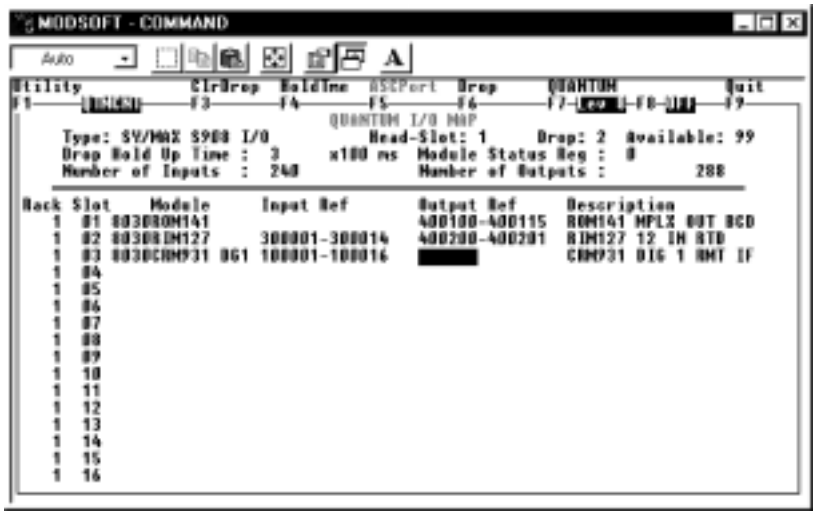


Figure 161 Quantum Remote I/O Map with Sy/Max

To change from a Quantum drop to SY/MAX or vice-versa, it is necessary to delete the drop and add a new one, with the desired drop type.

When Online: to distinguish between a SY/MAX drop and a standard Quantum drop, the SY/MAX module ID's must be unique (as stored in the controller I/O Map area), and thus not conflicting with other Quantum modules. This identification is accomplished by the module ID residing in a separate group of ID numbers, as 20xx hexidecimal, where the xx is the usual SY/MAX module's ID. Within the gcnftcop.sys file, the SY/MAX modules are grouped by module ID, and have a leading 'S' (e.g. S201E) instead of 'L' character (e.g. L002F) to set them apart from the usual Quantum modules.

There is a set of SY/MAX modules in the SY/MAX list obtained by pressing the **?** key:

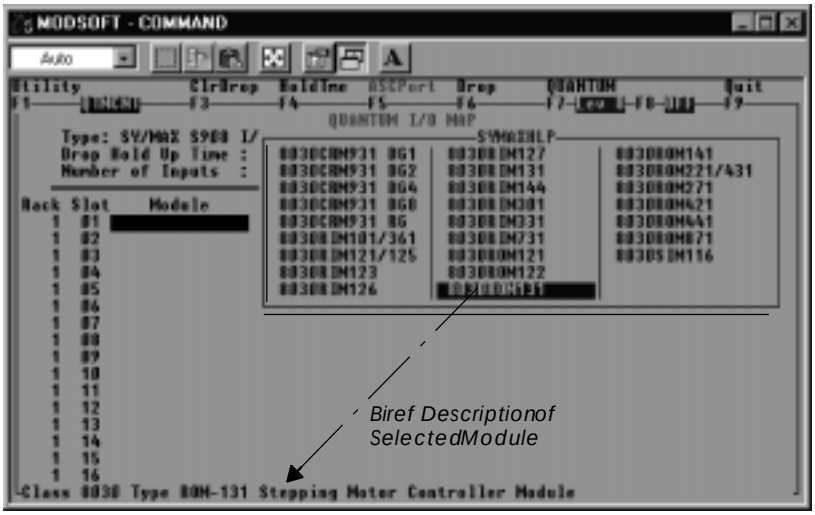


Figure 162 SY/MAX Module List

The fact that several modules have identical SY/MAX Code values requires that the module name and description for them be somewhat generic and that there be only a single entry in the file for each module ID number. All modules sharing the same ID will necessarily share identical I/O treatment from the controller. It will thus be impossible to see in a Modsoft screen the fact that a slot has an 8030RIM361 as opposed to 8030RIM101, which both share the same module ID number.

☞ **Note:** SYMAX drops are Quantum remote drops with special SYMAX modules installed and (or) I/O mapped. In online mode, when a SYMAX drop contains no modules and the I/O map is cleared, the "SYMAX" identity is lost, and the drop reverts to a Quantum drop. It is at that point impossible to change back to a SYMAX identification without returning to the Offline Mode and redefining at least One SYMAX module. Modsoft warns you if you are about to delete the last (or all) modules in a SYMAX drop while online.

5.10.1 Empty SY/MAX Drop particulars

There is nothing unique in the controller's I/O Map to distinguish SY/MAX drops from Quantum drops, except the identification of the modules you enter using Modsoft. If you delete all modules while *On-line* Modsoft will identify the drop as a Quantum remote drop. Also, if you were to attempt to delete the last module from a drop, a warning is displayed allowing you to abort the edit. A similar condition occurs when using the Clear Drop menu and a warning is issued and you can answer Y or N.

If you want to move the only module in a rack to another slot, you must create it in the new slot first then delete it from the original slot.

If a new SY/MAX drop is added while On-line, Modsoft considers it an illegal drop *until at least one module is placed in it.* If you attempt to start the controller before assigning at least one module in the newly created SY/MAX drop, a failure occurs with a 4000 error reported. It is necessary to maintain the “illegal drop status” until the first module is entered, otherwise it immediately reverts to a Quantum remote drop not allowing you the opportunity to enter a module.

If you create a SY/MAX drop by mistake and are On-line, and want to convert it to a Quantum remote drop, (without entering and deleting modules) use the Clear Drop function from the I/O Map menu. The warning will appear then continue with a yes answer to revert to Quantum remote I/O.

5.11 785E Quantum Support

Modsoft Configuration as detailed thus far in this chapter is a tutorial based on Modsoft alone but due to the wide range of hardware products supported, specific utilization of Modsoft utilities and functions are presented as examples. The 785E with Quantum support combines previously discussed functionality with specific requirements needed to run Quantum I/O modules from a 785E Controller. A summary of these requirements include:

- A Quantum formatted I/O Map is written for Quantum drops
- The 1K S908 supports Quantum I/O
- Changes in Configuration to accommodate either 800 or Quantum I/O

☞ **Note** A fully constructed I/O Map must not exceed a configured size of 14243 references.

5.11.1 Configuration Overview Screen Changes

You begin your configuration programming sequence at "Configuration Overview" where you select the `PLC Type` ie. 785E. The Execpack prompt box that allows you to select between 800 series I/O and Quantum looks like:

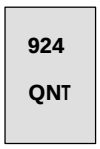


Figure 163 785E I/O Select Prompt

For 800 series I/O select the **924** then you are further prompted to select from either a "512 S908" or a "1K S908".

If you select Quantum I/O "**QNT**" the configuration parameters are updated to Quantum defaults of 1 segment and 512 I/O Map reserved words.

☞ **Note** 785E with Quantum IO goto drop is exactly like the 800 series goto drop.

There are two additional menu items available from the Drop function select entry. They are Add Drop and Delete Drop:

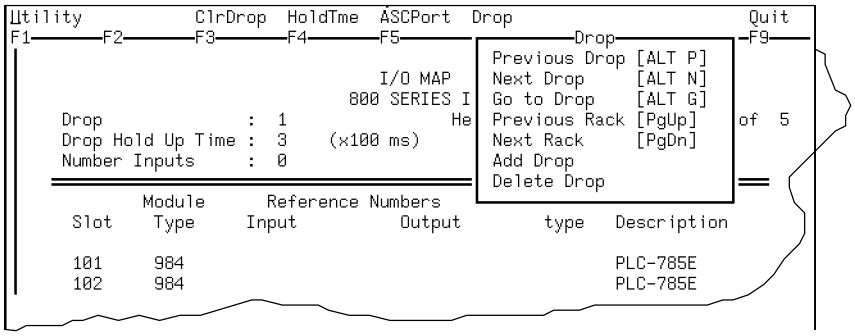


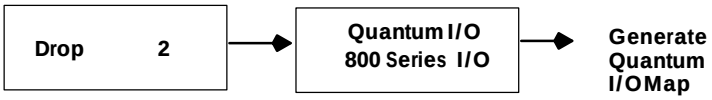
Figure 164 Add and Delete Drop

5.11.2 Add Drop

When you configure the PLC as a 785E with Quantum IO, AddDrop is allowed and functions exactly like the add drop for the Quantum series except that the head number is inserted by Modsoft and you only supply the drop number in the range of 2 to 16.

The Head number is shown here so that the user will not be confused by the head number in the I/O Map screens as well as in the Lister Cross reference for the I/O Map references.

After selecting the drop number, you are prompted to select the drop type, (exactly as in the quantum application).



The screen prompt bring you to a point where the Quantum I/O Map is displayed. Quantum I/O Mapping as previously defined is applied here.

5.11.3 Delete Drop

785E with Quantum I/O delete functions exactly like the full Quantum series delete drop.

5.12 VME I/O Map

The VME controller can be configured for either 800 series or Quantum I/O. The VME controller's environment dictated some specific rules for I/O Configuration such as:

- Drop 1 is not a local drop in the VME controller, it can be either a Quantum S908 or Distributed I/O. It is possible to have a VME I/O Map with no drops in the VME controller. Potentially, there could be no IO modules in a VME system, with a desire for no drops.

- When there are no drops, (VME 486 only), creating the first drop in the config overview screen will request the drop type from the user for drop 1. Deleting drop 1 in the VME 486 is allowed as well as deleting all drops in the VME 486.

You will see that the VME with Quantum I/O type an 800 series I/O map is shown but the "Drop" function pulldown is modified to include the selections:

Add Drop
Delete Drop

☞ **Note:** Modsoft shows non-health in the I/O Map itself with '*' chars adjacent to unhealthy modules when online and running.

5.13 Profibus Support

Configuration of the Profibus modules is not supported by Modsoft. Modsoft will provide only I/O Map, and help support for modules and zoom support for the ladder logic DX's.

5.13.1 DX Zoom for PROFIBUS Comm Loadables

Modsoft supports loading and providing zoom screens for the ladder logic comms DX loadables associated with them.

5.13.2 I/O Map Entries

Modsoft allows the modules to be included in the I/O Map, and help screens for the modules are provided. The modules do not use parameters for configuration.

5.13.3 Configuration

There may be loadables or config extensions to configure the modules. In either case, Modsoft is capable of uploading, downloading and/or installing the configuration loadable into the configuration. Modsoft is not expected to modify the configuration data contained in the loadables or Config Extension.

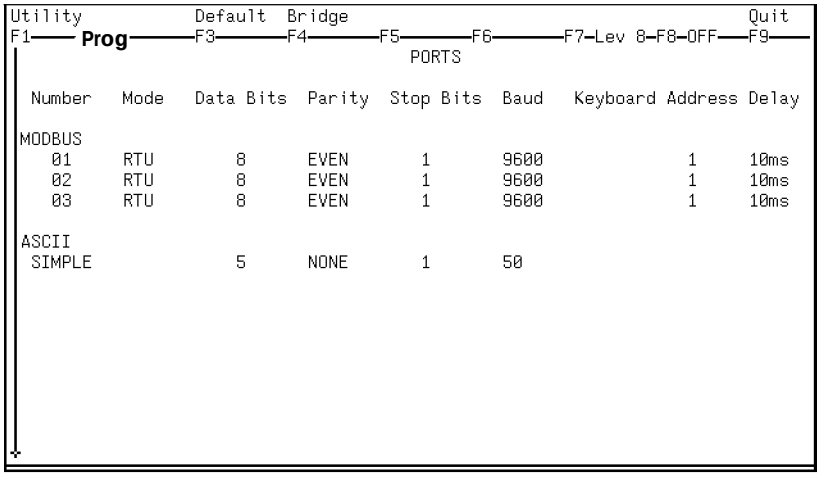
5.13.4 Profibus With M1

The Profibus supported M1 PLC requires an outside vendor's programming tool similar to the Quantum option. Modsoft does not provide this programming support.

5.14 Ports

"Ports" is a submenu entry of the main configuration menu. The screen is used to set the modes of communication for the Modbus ports, Simple ASCII and RTU ports.

Expert For installations that provide modules with Modbus ports, the Keyboard parameter is replaced with a Head-slot selection. The lower illustration would be typical if **Quantum** is the configured type. The input data is an address (required for NOM 211 or 212 modules for example) and can be between 0 and 16, where 0 is specific to the CPU no matter where it is installed.



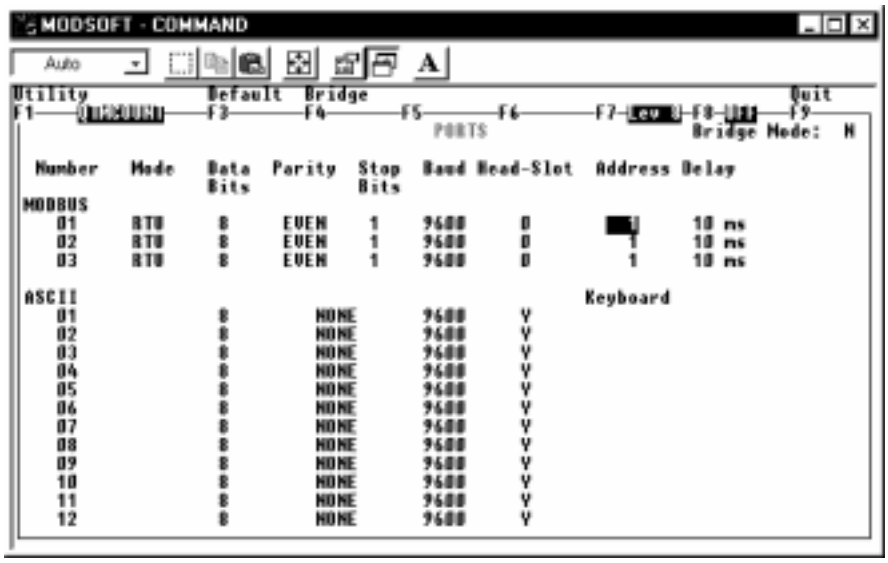


Figure 165 Modbus Port Selection Screens

5.14.1 Modbus Port

This function is available after selecting the “Ports”menu entry. You can select RTU or ASCII Mode. RTU (Remote Terminal Unit) mode is used to communicate between the controller and the personal computer. ASCII (American Standard Code for Information Interchange) mode is used for Modbus communication to various host devices. This selection sets the number of data bits, which cannot be edited. RTU corresponds to 8 data bits. ASCII corresponds to 7 data bits and each of the possible ports are available for selection. You can select 1 or 2 stop

bits (The Momentum M1 uses only 1 stop bit) and can select parity of: NONE, EVEN or ODD.

You can select from the following baud rates:

50	75	110
150	300	600
1200	1800	2000
2400	3600	4800
7200	9600	19200

The default value for baud rate is 9600.

You can also set the programmable controller address. This value must be within the range of 1 to 247. The default address is 1.

You can input a delay-time of 1 to 20, representing multiples of 10 milliseconds (ranging from 10 to 200 ms). A typical setting of 1 (10 ms) is the default delay time. The delay time is the PLC Modbus port turnaround time. If there is a near zero time delay in the PLC between receiving a message and returning a response to the host, it is possible that the response to a message will be lost because the host device is not ready to receive. Model “E” controllers allow up to 1 second of delay time.

☞ **Note** Delay is only valid on 984A, B, X, E and Quantum Model Controllers

The special Controller types that provide Simple ASCII Port are 984A and 984B. The values to enter are:

Parity, Stop Bits, Data Bits, Baud rate.

The allowed values for Data Bits are seven and eight depending on the Mode selected. The default values for RTU mode are:

Even Parity, 1 Stop Bit, 8 Data Bits, 9600 Baud.

default values for ASCII mode are:

Even Parity, 1 Stop Bit, 7 Data Bits, 9600 Baud.

5.14.1.1 Default

This selection will set all selected port parameters to to preset initialized values. The default operating values: RTU mode, EVEN parity, 1 stop bit, 9600 baud, address 1, delay 1. This function is also available to return ASCII ports to an initialized parameter set.

5.14.1.2 ASCII Ports

This function is selected from the Mode select entry of the menu item "Ports". When you want to change between ASCII and RTU you place the cursor on the Mode field and press the **?** key or just press the return key while the cursor is on the Mode field The number of ASCII ports is specified in the configuration overview. You can enter the values for:

Parity, Stop Bits, Data Bits, Baudrate, Keyboard/Non Keyboard.

A corresponding selection list opens automatically when entering an entry field. Within the line the cursor can be moved via <**Cursor Right**>, <**Cursor Left**> . There is a menu entry to set all parameters to default values, just as in the Modbus port selection. The allowed values for Data Bits are 5 through 8. The default values are:

Even Parity, 1 Stop Bit, 7 Data Bits, 9600 Baud, Keyboard.

The definition of the parameters is accomplished similar to the Modbus Port Settings.

5.14.1.3 Bridge

When networking controllers, a Modbus device connected to port 1 can communicate with the controller it is connected to as well as log into any nodes on the ***Modbus Plus*** network. Your software Bridge selection is effective, as a memory parameter, for the Modicon "E" series, Compact 140, 145, 245 or 255 and Quantum Series controllers and activates the Bridge mode if set by pressing "Y". The Controller Communications parameter select switch can only be set to **Memory** for this option(19.2k baud) (See Controller Planning and Installation Guide for switch locations and labeling details).

The controller hardware defaults to bridge mode when the Mem/Dip (785E) is set to Dip or when the Quantum-CPU three position switch is set to the middle (RTU Mode).

5.15 Segment Scheduler

The segment scheduler available from the configuration main menu lets you select and set the Constant Sweep, Watchdog Timer and LL Edit Timeslice. A detailed discussion is available in the:

Modicon 984 Programmable Controller Systems Manual GM-0984-SYS.

The following parameters for Initialized and Inserted Schedule Numbers can be configured:

Type - Reference Number - Sense - Program Segment -Drop Input - Drop Output

An example screen for the segment scheduler is:

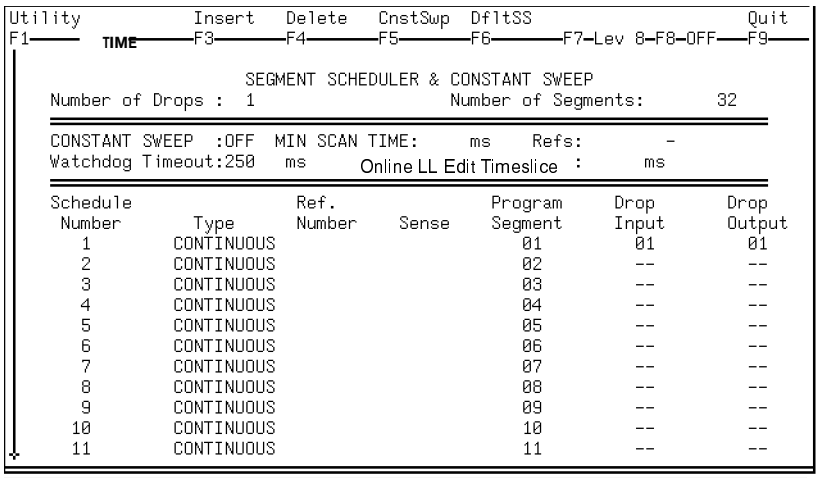


Figure 166 Segment Scheduler Example

The number of drops is displayed in the header of the table. Segments can be inserted or deleted by menu functions. With the cursor on a Type, you can press the ↵ key to display a type select list. After inserting a “Type” entry, continue to fill in the data by moving the cursor to the segment number and input and output drops. Initially the new entry is defined in CONTINUOUS mode. You can only set Reference and Sense values if the **Type** field is set to CONTROLLED mode is the “Type” selected. Only 0XXXX and 1XXXX are valid reference numbers. Possible entries to the segment scheduler are: WDT Reset (Watch Dog Timer), CONTINUOUS, CONTROLLED, and EOL (end of logic). The mode can be selected in a pulldown menu. There can be no empty lines between segments nor can you set a WDT in the first line

or set WDT's in consecutive lines. If you select EOL, a warning is displayed that all following lines are not written to the configuration.

5.15.1 Insert / Delete

The editing menu select functions allow you to Insert or Delete segments in the schedule as well as Set the Constant Sweep toggle between ON and OFF. If you have made changes and want to revert to a known schedule you can select "DfltSS" to obtain default values.

Caution If a complete segment is deleted any existing comments related to the networks in that segment remain in the internal database.

If you reinstate that segment and use it for other networks the old data base text is assigned to the new network numbers.

Tip If deleting a segment, complete the delete by eliminating the associated comments. Proceed as follows:

Export the comments and close Modsoft
Use an editor to revise the comment data base.
Import the edited files.

5.15.1.1 Reference Number

When a segment is set to CONTROLLED mode, this field prompts for a 0XXXX or 1XXXX reference number. This reference is used to decide when the logic for that segment is solved. The sense field can identify whether the contact should be ON or OFF in order to solve the logic for that segment (available only if CONTROLLED mode is selected for this line).

The segment field lets you enter the segment number. Valid entries are 1 through 32. The Input and Output fields let you enter the drop numbers for input and output. It is possible to select a zero drop number. No input or output is serviced in this case.

5.15.1.2 Constant Sweep

Constant sweep Functions ("CnstSwp") are provided to turn ON or OFF the constant sweep function. A target scan time can be entered in multiples of 10 milliseconds (10-200 ms). Additionally, you must specify the first of two 4XXXX registers that are used to store the selected and the actual sweep time. After Constant Sweep is configured, the controller runs with the defined constant scan (sweep) time when the controller is in the RUN mode.

Each scan starts only after the expiration of a constant period of time, but only if the actual

scan time is less than the desired scan time. If the actual scan time is greater, the setting is ignored. The setting in reality, is a Minimum Scan time.

Note This function can not be used to speed up the execution of the programmed logic or to terminate a scan early.

Also, this function uses the next consecutive register which contains the actual scan time, when the actual time exceeds the preset value.

When this function is used in conjunction with the watchdog timer, the watchdog value should be greater than the Constant Sweep value.

5.15.1.3 WatchdogTimeout

The Watchdog Timeout sets the maximum scantime allowed for logic solve. If the maximum scantime is exceeded, the Controller stops. The Watchdog Timeout range is 10 to 2550 ms. in steps of 10ms.

5.15.1.4 LL Edit Timeslice

The Online LL Edit Timeslice is the multi-scan timeslice, The value assigned sets the maximum amount of time per scan that is allowed for processing Edit updates to Ladder Logic functions such as Insert/Delete and Search. This value impacts scan time directly: a low value has less impact in increasing the scan time but Ladder Logic edit updates take longer. The range is 3 to 99 ms. If you increase the timeslice from 20 to 30 ms the scan time may increase by 10ms during editing.

5.15.1.5 Default

The segment scheduler is created by default with all defined segments continuously controlled and with I/O drop Numbers equal to Segment Numbers up to the number of drops.

The DfltSS is used to return to the value established on the overview configuration selection. This also works as a reset if you have altered the controller type selection that has a different segment value. A System Message will make you aware of this condition.

5.16 Loadable

Loadables are programs developed to provide specific control functions beyond standard Ladder Logic but work in the Logic as a standard element.

The Loadable selection of the configuration editor, offers the standard menu entries Utility and Quit as well as the Dir Edit Loadable specific functions:

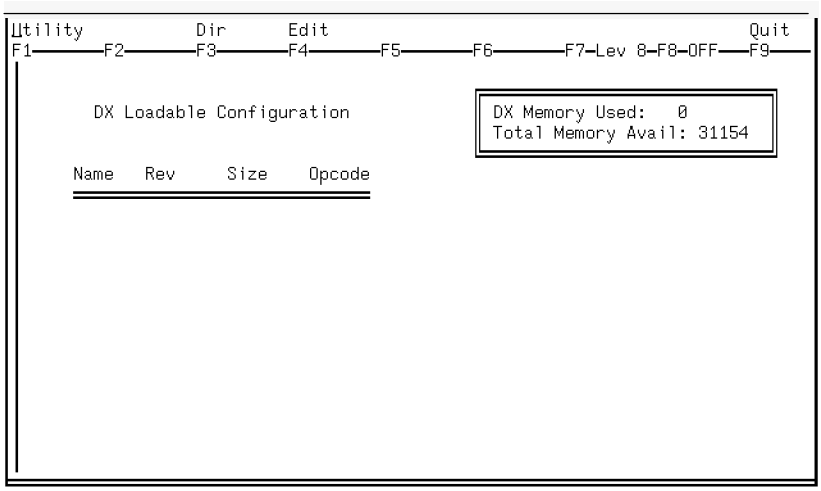


Figure 167 Dx Selection Screen

☞ **Note:** Loadables that have an extension of .exe, produce different library files for 16 bit controllers (e.g. Quantum 186, Compact, 984 slot controllers etc.) and 32 bit controllers (Quantum 424, VME 424). The loadable files are only common in the original distributed file (relocatable format). Due to this difference, it is not possible to load and run a library loadable file for the 186 on a 486 controller and vice versa. It is therefore STRONGLY recommended that you segregate your project programs for 32 bit controllers from all other project files.

5.16.1 Dir

The pulldown under the Directory function allows you to elect a Load or Delete operation from the file system.

5.16.1.1 Load

Loadables with .DAT or .EXE filename extension are loaded and have their filenames posted to the Loadable selection list. If a .DAT contains multi-file Loadables each loadable will be separated into an independent Loadable. The file characteristics produce a Loadable name, revision number and family type. The Loadable files are placed in the Program directory. If this path is left at default it is c:\modsoft\programs\ . Modicon State Language in the form MSL.exe may also be loaded. The Loadable format includes:

Loadable Name	Rev	Type	.DAT an example of this format is:
HSBY	102	8	.DAT a hot standby loadable rev 1.02 for -8 controllers
HSBY	000	A	the type of A is for 984 A/B/X

in practice the example has no tabs i.e.

HSBY1028.DAT
HSBY000A.DAT

User Loadables are similar to Schneider Automation produced loadables but the Customer can design their own application for inclusion in a Ladder Logic program. User Loadables have .EXE extensions. There is one .EXE per user loadable. Refer to GM-CLSS-001 for design instructions. User loadables for a given program are gathered together and have the file extension .USL associated with them

As the I/O module offering expands you can enter a module specific driver just as adding a loadable Dx. The example below illustrates your response to the "Load" selection with the specified path and file name to *loadfrom* the A: drive. Your selection is displayed as below:

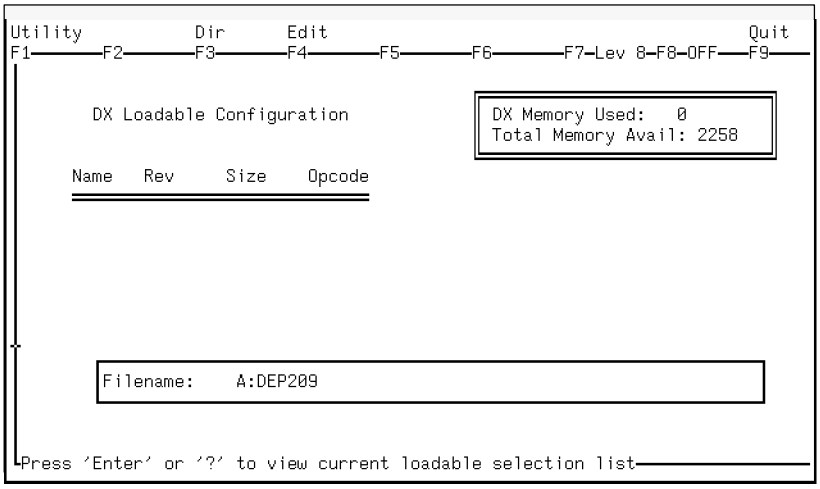


Figure 168 Using the Load Function to ADD Loadables

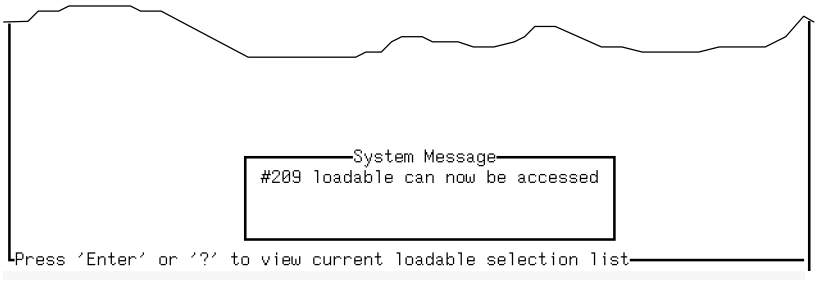


Figure 169 Display Verification for the Current Loadable

The Loadable I/O is identified by the # symbol, in the configuration selected display. You can display the above example I/O loadable in the Library of loadables, by pressing the ? key

5.16.1.2 Delete

When you select the Delete option you are prompted for the file.DAT to delete. A warning is posted if you try to delete a Loadable currently Configured. The proper sequence is to delete the Loadables (from the Loadable configuration) ***then*** delete the file.DAT. This function is also applicable to the user loadable .EXE file.

☞ **Note** This function will Delete the files from your fixed or removable disk media.

5.16.2 Edit

The Edit pulldown submenu allows you to select from Three Edit Functions:

InsertDeleteOpcode

The Quick Key to select each respectively is <**Ins**>, <**Del**> and < **=** >.

5.16.2.1 Insert

When selected, the cursor is moved to the blank line on the Loadables display and the selection list of Loadables loaded into Modsoft is displayed (if any). You make the selection of the Loadable desired from the displayed list.

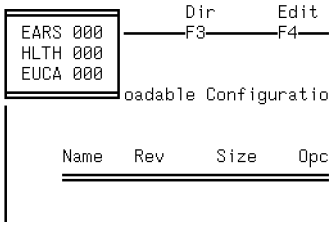


Figure 170 Loadable Select List Example

If the opcode for the selected Loadable has been previously assigned, a list of available opcodes is displayed and you must choose one from this list. The Loadable selection list can also be displayed by pressing *Enter (↲) or the < ? >* key while the cursor is on a blank line, on the Loadable screen. When you have selected a Loadable function from the displayed list the one selected will no longer be available in the displayed list and has a * preceding the Loadables in the selection list. The Loadable list also displays the revision level of the Software. The above example displays the revision level of Zero (000). The next revision is 001 etc,.

☞ **Note** If only one Loadable is loaded in Modsoft, there will not be a select list displayed (as above). It is directly put in the “Selected table”

☞ **Note** If two Loadables of the same name but different revisions are loaded into Modsoft and one is selected, both are not accessible and have a * preceding the Loadables in the selection list.

5.16.2.2 Delete

Placing the cursor on a Loadable and pressing the return key deletes the loadable name from the configuration. The Del key can also be used. When the loadable is deleted from the configuration, it is available to be selected again in the loadable select list.

5.16.2.3 Opcode

The Opcode selection list is displayed and allows you to change the opcode of the Loadable on which the cursor resides. This action causes the .DAT/.EXE file, in the Program directory, to be updated to incorporate the change (the = key is the Hot key to bring up the Opcode list).

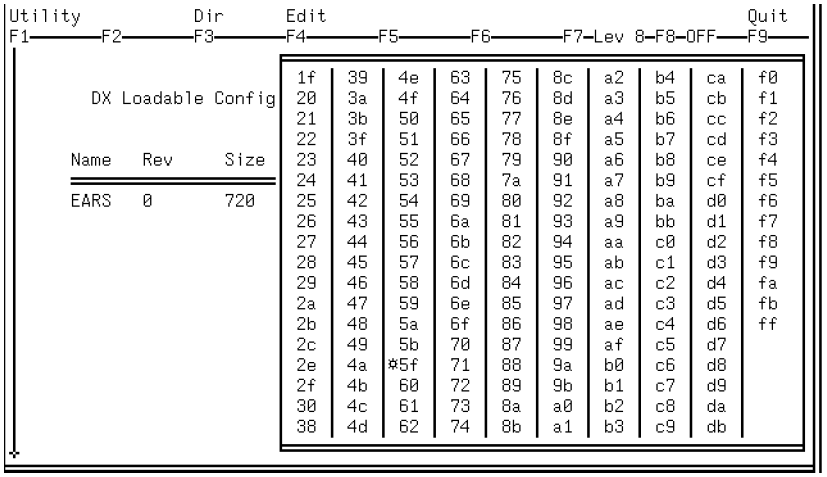


Figure 171 Example of Loadables OpcodeSelect List

Note When doing a PLC to File Transfer and .DAT/.EXE is not in the Program directory, it will be created, but tracking help is not available for the Loadable.

If the Loadable Opcode in the PLC is different from the Opcode in the .DAT/.EXE file in the Program file, the Opcode in the file is updated, When the .DAT/.EXE file contains different code from the Loadable in the PLC, the user is prompted to delete or rename the .DAT/.EXE in the Program directory.

Warning *SWAPPING LOADABLES ASSIGNED OP CODES*, after having used either of the loadable function blocks in the Ladder Logic and transferring the program to the PLC, unpredictable execution in the PLC can occur.

The CONDITION:
Changing OP CODES after the function block is used in the ladder program does not automatically change the OP CODES in the Ladder Logic to match the new op code. If a user simply changes the op code of the panel program, the ladder logic will not download since its op code (the old one) does not exist. If after changing one loadables op Code, another is either changed or installed as well, and it is assigned the same op code that the first one used to have, this constitutes swapping op codes, and Modsoft no longer reports an error on downloads, since each function block in the ladder matches to an actual loadable.
The ISSUE:
Once in the controller, the function block(s) execute and produce the wrong effects, like as if ADD and MULT were swapped everywhere in the program. The effects are virtually unpredictable.

5.16.3 Loadables Disk

The delivered software contains 3 disks in addition to the Modsoft set. This additional disk contains a number of Loadable functions, Loadable executives and device drivers for A120 I/O. You can put the software routines in modsoft directories or use them as required directly from your **a:** drive using the import (Load) facility of the `Configuration → Loadable` Selection.

5.16.3.1 Loadable Dx Summary

There are two ways to put a loadable into your logic programming.

1. Select it from the existing (Configured) library of available elements.
2. Add to the list using the configuration `Loadable` function provided.

   a) Use ***Dir*** to define the location of the file to input to the list of loadables.

   b) Use ***Edit*** to `Insert` the selected (from the list) loadable into the program Configuration.

5.17 Configuration Extension

The hardware and software implementation of Configuration Extension block functionality requires the Controller application programmer to be familiar with supporting System and Communications details. You should refer to:

Modicon Ladder Logic Block Library 840 USE 101 00
Modbus Plus Network Planning and Installation Guide 890 USE 100 00

5.17.1 Configuration Extension Menu

The Cfg Extentry on the Configuration menu line provides a sub menu and input screen that allows the standard Utilityselections, the configuration extension block selection, block parameters (including memory requirements) and Zoom function.

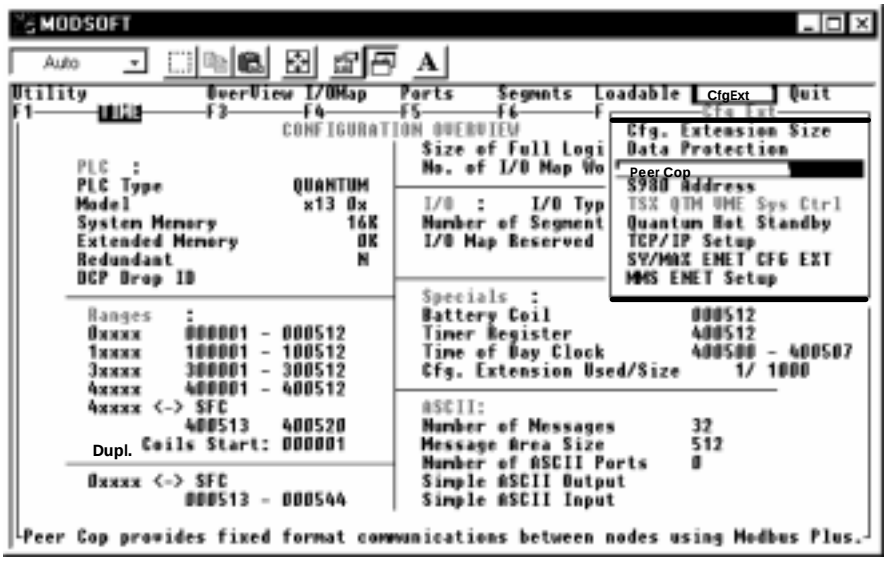


Figure 172 Initial Screen to Select Function

5.17.2 ExtSize

Extension blocks ***require*** memory resources that decrease the amount of user memory available. Unlike the existing configuration table, the configuration extension does not have a fixed size or address. *You must start the creation process by assigning a memory size.* The Menu selection "ExtSize" is used. When selected the cursor is placed on the Total Words field and

you can set any value from 0 to a maximum of 32768 words (depending on controller memory size). The value you select is displayed and the amount you use is calculated for reference.

If you attempt to reduce the size below that required for the currently used extensions, you receive a warning and an opportunity to change your mind about the new size. If you allow the smaller size, all current configuration extensions are deleted. If you wish to delete only a selected configuration extension, do so via the configuration extension's individual delete feature.

To post the available Configuration Extension functions to the initial screen you select the menu item `Cfg Ext` and see a selection list.

Cfg. Extension Size
Data Protection
Peer Cop
S980 Address
TSX QTM VME Sys Ctrl
Quantum Hot Standby
TCP/IP Setup
SY/MAX ENET CFG EXT
MMS ENET Setup...

If you have assigned a size to the configuration extension, and configured one or more configuration extension functions, then choose to reset the size to zero, you are warned that any existing Configuration Extensions will be deleted. This message is the same whether there are one or 10 Configuration Extensions that will be deleted. You have the option of escaping, without setting the size to zero by entering "N" to the Y/N choice.

If you have reserved too little memory, you are prevented from entering certain new configuration extensions. Some others like Peer Cop and Ethernet setup, use variable amounts of memory. If you use too much, you can not save your immediate changes. Be sure to allocate more than needed, then adjust it downward as your program is completed.

5.17.3 Data Protection

The data access protection function provides you with the ability to configure the controller for controlled access over part of the user state RAM area. With this function, a Modbus Communication write access range can be setup for some of the reference types supported while references outside this range are read only.

A Data Protection Block can only be used if the PLC supports configuration extension. If not supported, an error message is displayed when you try to download the configuration (unless suppressed by use of the User Preference File).

Your initial entry is to a display that lists currently defined Configuration Extension blocks. To create the new Data Protection Block press <**Y**>. You are informed that it is not configured and asked if you want to create it. Your answer is either Y or N. The cursor is then positioned on the start field of the 0X reference.

> **Note** If you forget to set the ExtSize you will not be able to enter any of the configuration extension blocks

The software calculates the ranges of 0X and 4X reference blocks from your general configuration reference “Range” definition and initializes the area to write access as:

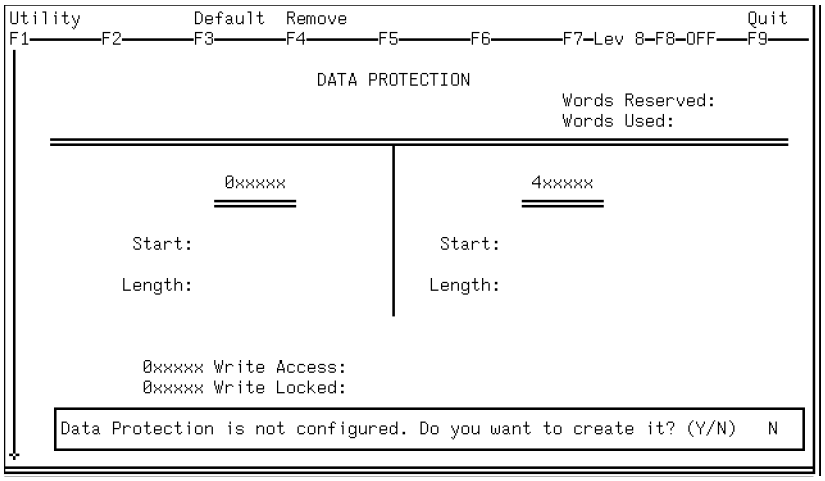


Figure 173 Initialized Parameter Entry Screen

If you wish to Remove the Data Protection function, press the <**F4**> key.

Press <**Tab**> to elect the Default “To Protect” which records the parameters.

<**Esc** > returns to the Configuration menu.

A “1” in the length field of the 0X parameter field is for a single coil not 16 coils.
A “1” in the length field of the 4X parameter field indicates 1 register.

To lock an area within the bounds of the defined write access area you have to specify *where* and *howmany* 0X and 4X references are to *remainaccessible* You do this by entering the Start and Length field data. In the example, illustrated, the 0X references 1 through 500 as

write accessible and the balance are locked (can only be read). For the 4X the first 512 registers are accessible and the amount Locked has not yet been set.

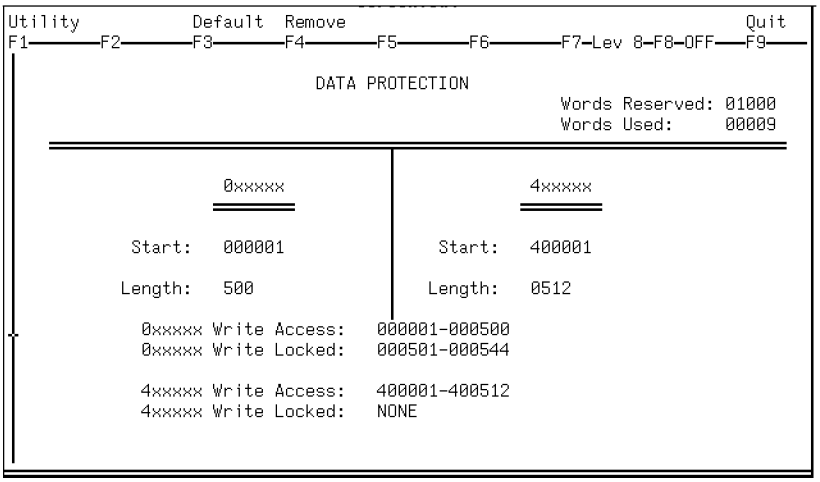
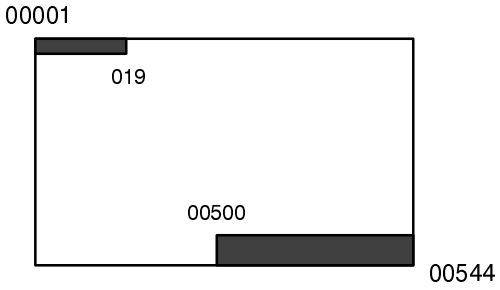


Figure 174 Result Calculation and display example

You can protect an area at the beginning and at the end of the memory within the defined total block. For instance; if in the above example you wanted to protect the memory 00001 to 00019 (20 locations) as well as 00500 to 00544, i.e. a block that looks like:



You could enter a Start of 20 and a Length of 480 (the difference between the 20 and 500). The display identifies the locked areas as:

☞ **Note:** Non Modsoft writes such as controller driven Peer Cop writes, are not precluded from accessing this area of memory.

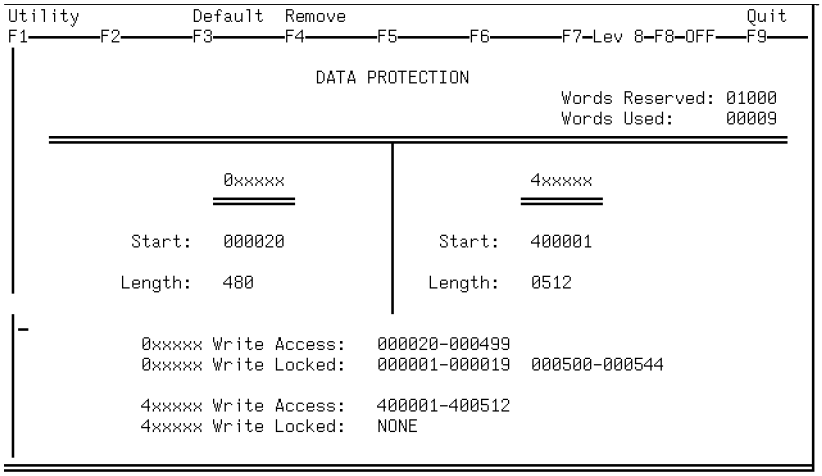


Figure 175 Multiple Locked Areas

5.17.4 Peer Cop

The Peer Cop configuration extension allows you to configure certain continuous, fixed format communications between the controller (in which it is defined) and all other nodes on ***the same Modbus Plus link.*** Peer Cop requires Modbus Plus and the 984 series controllers including E models, Compact 'E' series, Magnum and Quantum Series. Due to some differences in architecture in the Quantum Series product, when Quantum is the configured controller the Peer Cop screens are slightly different. In particular the head number is displayed and entered as a slot number between 1 and 16 instead of the link number.

Each Peer Cop configured communication specifies a source data block. The source data block is of fixed location and length and is continuously moved, to a fixed destination data block. This data transfer type is useful for transferring state information between controllers and for communicating with distributed I/O nodes on Modbus Plus.

Peer Cop communication is not appropriate for sequence dependent communication that must be performed exactly once. The standard MSTR element is used for those logic dependent requirements with certain restrictions.

☞ **Note** *Peer Copped data transfers cannot pass through bridges or bridge multiplexer modes.*

Like the I/O Map, the Peer Cop can only be configured with the controller **stopped.** Once the PLC is configured and started, the transfers are performed automatically.

The implementation and application of Peer Cop data communications in a Modbus Plus environment is documented in 890 USE 100 00. This Modsoft User Guide deals *only* with Modsoft panel software and the required user interface.

A menu item in the Peer Cop is available to delete the current node on the screen. A warning is given and the node is deleted if (Y) is answered. If the last node is deleted, a window opens to allow entry of a node. This condition is identical to the initial screen of an empty Peer Cop.

5.17.4.1 MMI-PeerSystem Interaction

As you use Modsoft to program or monitor a control program with Peer Cop implementation, you should be aware of the following points that are detailed in the previously referenced documentation:

- 984 Stop Code
  The controller stop code word, maintained in the configuration table is updated by the executive if an invalid block status is detected. The system message 0001 indicates a Bad Configuration.

- MSTR Error 200B
  This Error is put into the MSTR status register to indicate a conflict on Read and Write Global data. This conflict is an *either/or* situation: **if** you use the Peer Cop to issue a data stream or set the receive conditions **then** you can not also use MSTR to perform global data transfers.

- Peer / MSTR timing
  Peer and MSTR impact scan rate in different ways. Peer Cop transfer is conditioned by the Modbus Plus network and its token rotation time. MSTR is conditioned by the internal controller rates and can take many scans to complete one transaction.

☞ **Note:** Non Modsoft writes such as controller driven Peer Cop writes, are not precluded from accessing any area of memory.

5.17.4.2 Storage Requirements

Before selecting Peer Cop from the `Cfg Ext` pulldown list you must use `ExtSize` to set the memory storage requirements.

☞ **Note** The remainder of the Cfg Ext pulldown functions remain disabled until the ExtSize is set.

There are four types of Peer Cop requests:

Global data **input**
Global data **output**
Specific data **input**
Specific data **output**

Depending on your requirements you can estimate the memory needed where:

- There is always a 9 word overhead.
- If Global Output is configured add a fixed 5 words.
- If Global Input is configured then add 1 + 2 times the number of sub-entries (8 maximum) for each device entry (number of nodes on the network) (64 maximum).
  Maximum is 64 x (1 + 2 x 8) = 1088 words
- If Specific Output is configured then add 2 words for each device entry (64 maximum. Maximum is 64 x 2 = 128 words).
- If Specific Input is configured then add 2 words for each device entry (64 maximum). Maximum is 64 x 2 = 128 words.
  Based on the above, the minimum size Peer Cop could be 20 words while the maximum could be 1366 words for each link, and you can have up to 3 links.

5.17.4.3 Peer Link Topology

Depending on the way you have implemented the Modbus Plus physical media, you may use the "AddNode" and "GoToNode" commands to create Peer Cop data communications templates for up to 64 Nodes on each of up to 3 separate Modbus Plus network Links. If dealing with a Quantum Series installation, the equivalent of *up to 3 Heads* can be configured.

The Link Number specifies the link that the described data applies to. A value of 1 refers to the "Built in" Modbus Plus Link. A value of 2 is for a S985 Option 1 and a value of 3 is for another S985 option 2. The ***conceptof multiple link*** Peer Cop Configuration is illustrated on the next page. The S985 link is designated by an internal switch setting at installation time. Refer to the hardware manual (GM-S985-001) as required.

Dealing with Quantum Series Heads is quite similar to other Modicon controllers, but the head number you give is a direct relationship to the backplane slot the head is installed in. A head end function can be performed by a PLC, NOE or NOM.

In this simplified example, you use the Link functions to define the data flow on link 1 then

create an ADD link to define the activity desired on link 2. The Modsoft Panel Application is used to:

Step 1 Write a Configuration including Configuration extension in controller Device # 4. The data generated in this controller is defined for Peer Cop using Link 1. For example, a Specific Output of 32 registers to Device 24 may be defined.

Step 2 Go to Device #1 and do the same thing as step 1 using Device address of #1 and a Link 2 Data definition.

Step 3 Configure Device #24 to receive Link 1 data from Device 4 and Link 2 data from device 1

Result If your registers have data which varies *and the register input in device 24 separates the 2 input areas*, you can monitor the change in register contents for each Link, using the Modsoft Reference Data display.

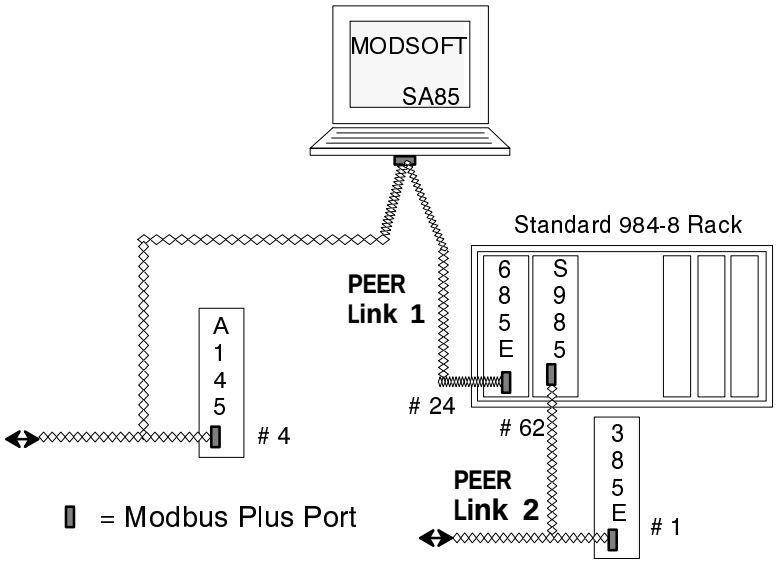


Figure 176 Modbus Plus PEER Link Example (without Quantum)

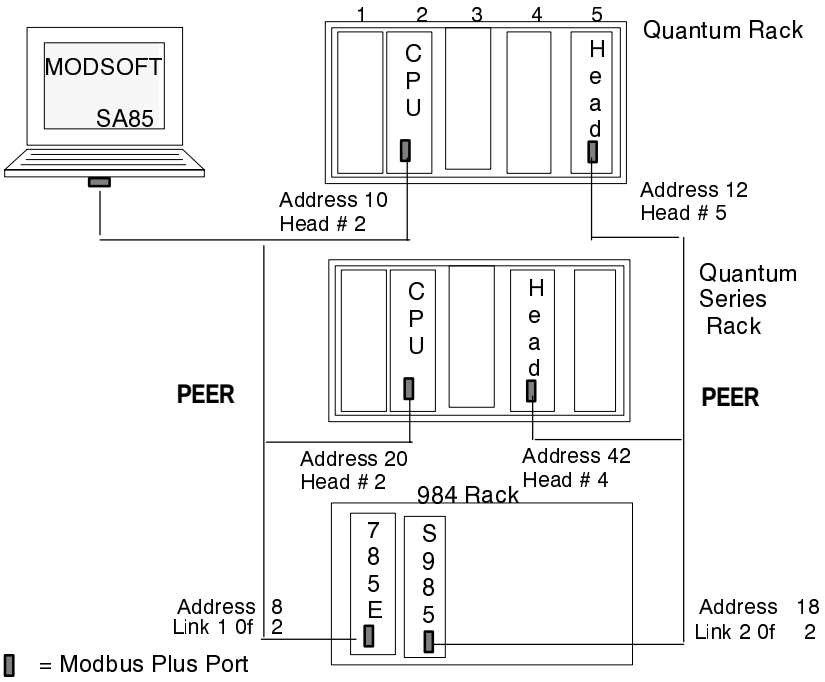


Figure 177 Example of Modbus Plus Peer Network WITH Quantum

5.17.4.4 Specific Input/Output Configuration

The default screen for Peer Cop entry is labeled "`PEER COP`" and is illustrated in Figure 178 and 180 as an example of default types. The screen is a data entry template comprising all four data types and providing a summary of settings that apply to the specific link/node as well as timeout, error handling and statement of memory words used.

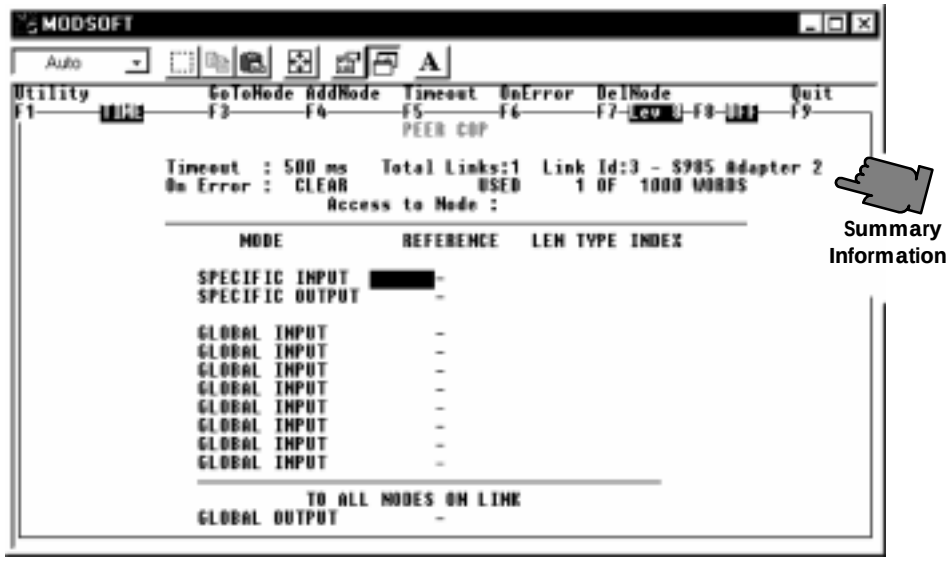


Figure 179 Initial Peer Cop Entry Screen (785E)

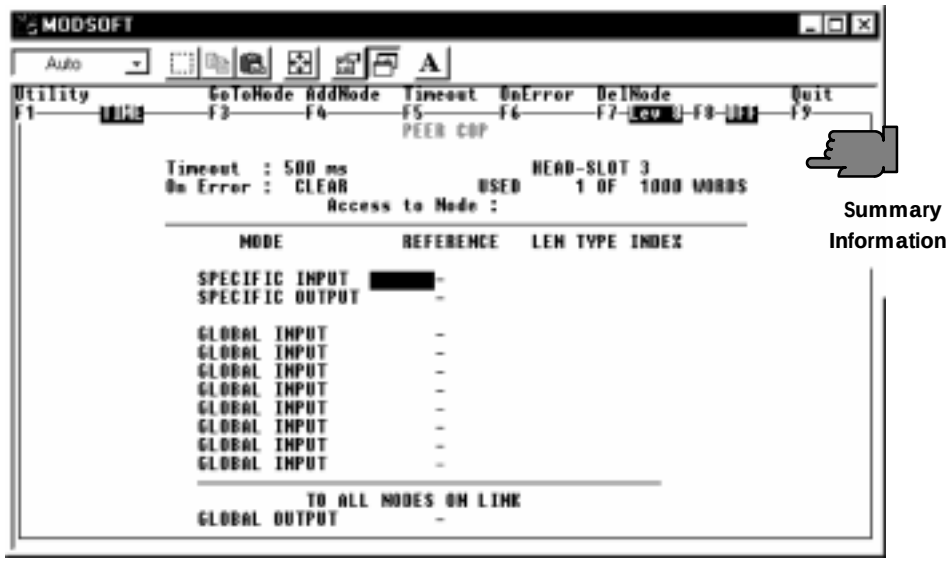


Figure 180 Initial Entry With Quantum Configured

The cursor is initially in the Link *(or Head)* field. If you are not editing an initial template you may press the Esc key which re-positions the cursor to the SPECIFIC INPUT field. To traverse the Links and Nodes you can re-display the Add Node select box from the main Menu line.

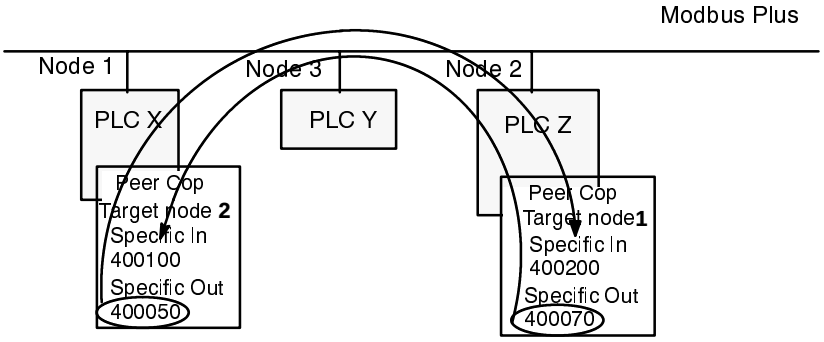
You can define all specific output blocks, sent from this controller, to the specified other stations on the link. The source of each block is a contiguous region of 0x, 1x, 3x or 4x state RAM, which varies from 1 to 32 words in length. If discretes are used, they must start on a word boundary (00001, 00017, 00033, etc).

The Type default is put in by the controller. Where different types can be specified, you make the entry from a display list displayed by keying the return key while the cursor is on the TYPE field.

You can specify all Specific Input blocks, sent to this controller, from the specified other stations on the link. The source of each block is a contiguous region of 0x, 1x, 3x or 4x state RAM, which varies from 1 to 32 words in length. If discretes are used, they must start on a word boundary (00001, 00017, 00033, etc). Output data from another node is accepted as input data by this controller ***only if:***

- The Peer Cop in the "Input" controller has a configured input block for the sending node (Source) and the length is an **EXACT** match between the two.

  For example:



☞ **Note** Configuring Specific input/output in a controller (device) does not automatically mean that the destination devices will receive and or process it. *The destination device must be configured as a sender to, or receiver from the source device to enable the process.*

The next illustration provides an example of the Length, Source and Type data fields. The following illustrates the above example completely filling in the template fields:

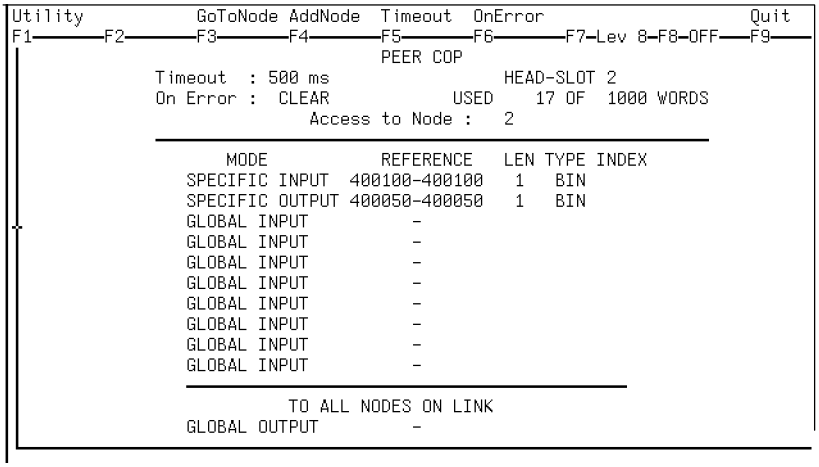


Figure 181 Peer Cop Specific I/O with Example Data Entry With Quantum

You can quickly and directly access any node template by using the `GoToNode` command. The next figure illustrates the template for node 64. Note; the previous template for node 2 specified “Global Output References”, therefore all nodes on this link will have those references available for control programming.

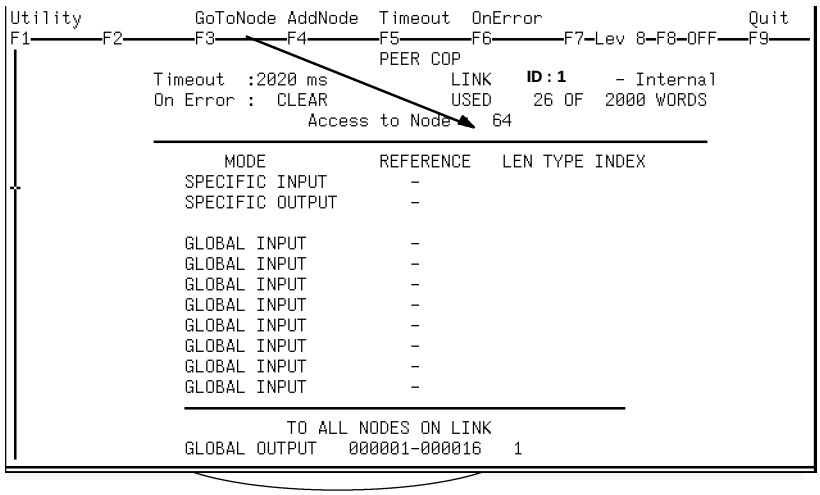


Figure 182 Template with Device 64 Address and Global Definition

The template difference for the E series controller is demonstrated in the following screens:

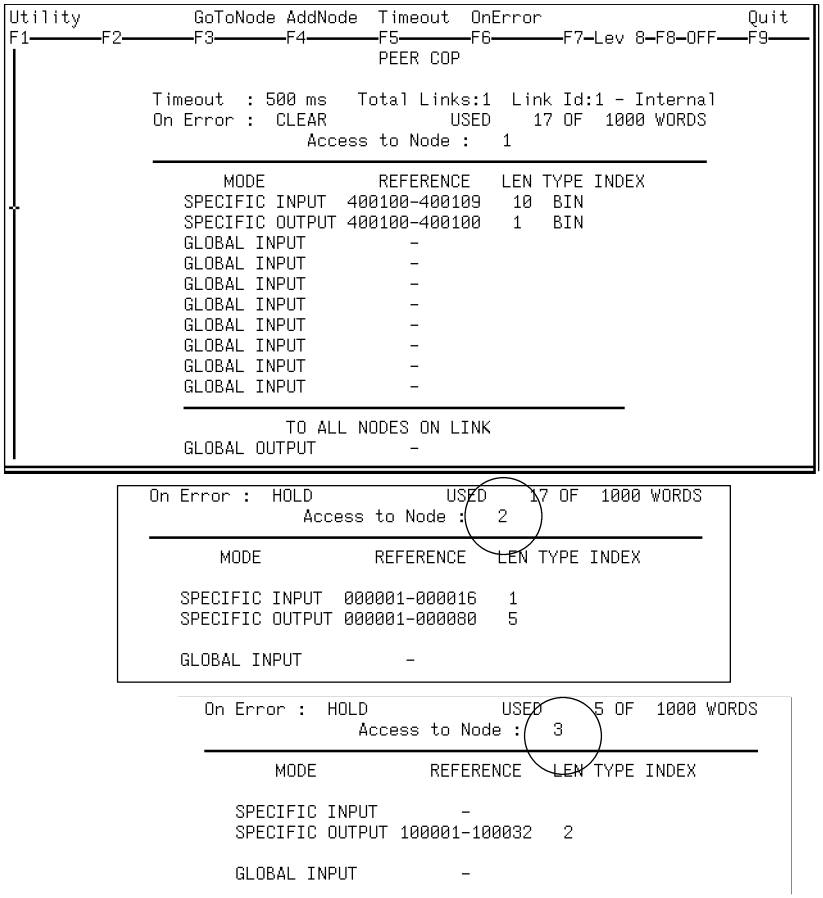


Figure 183 Peer Cop Specific I/O with Example Data Entry (E Series)

5.17.4.5 Specific Input - Output Summary

Given the proper configuration, if you complete the above template for *node2* with a specific input of 000001 length of 1 and Specific output 000001 Length of 5 and add a Specific output for *Node3* of 100001 length of 2 will result in:

sending 10 words to the device on Node 1
sending 80 discretes (5 words) to the device on Node 2
sending 32 discretes (2 words) to the device on Node 3 and

You are also configured to:

receive 10 words from the device on Node 1 and
receive 16 discretes (1 word) from the device on Node 2

5.17.4.6 Other Menu Selectable Support Functions

In addition to the Standard Utility Menu line entry, you have Peer Cop related functions available from the `GoToNode, AddNode, Timeout` and `OnError` entries.

- `GoToNode`
  Displays the Peer Cop menu that allows you to configure that node . This function has a pulldown:

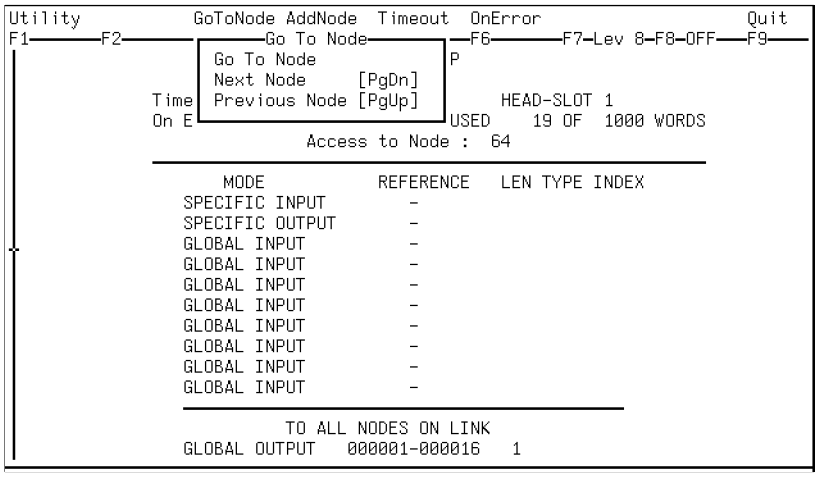


Figure 184 GoToNode Pulldown

If you select `Go To Node` and the node number you enter is not found you are asked if it should be created for you.

You can also traverse the node structure using the **PgUp** and **PgDn**

- `AddNode`
  This is very similar to GoToNode in that you select the Link and Node number that you want to add parameters for.

- `Timeout`
  This field allows you to specify a value for the Health Time-out interval. The default value is 500 Milliseconds. You can change it to any value in the range of 20ms. to 2 seconds. The value you use specifies the minimum time period that a Peer Cop configured communication must fail before the associated health bit is cleared. You should choose values in 20 ms. increments to account for implementation latency i.e., the configured time plus the time to assure the health bit is cleared.

  For example if your choice is 60 ms. the health bit is cleared no sooner than 60 ms. and no later than 79 ms. after communication has been lost.

- `Health Bits`
  There is a health bit for each Peer Coppednode. If Peer Cop Data is successfully communicated within the set timeout, the associated bit is set to 1. Otherwise, it is set to 0 and all data associated with that group is cleared (to 0). You must use the MSTR element with proper sub-function code (0009) to retrieve the peer cop health information. (See Modicon 984 Programmable Controller Systems Manual GM-0984-SYS).

Note All configured Specific output health bits are initialized to 1 for the first few scans to allow complete synchronization between controller, health bit time factor and line latency.

- `OnError`
  You have the choice of Clearing (CLEAR) the last set of received values or retaining the last set of received values (HOLD) if any error is detected.

- `DelNode`
  Once Deleted, you can re-enter node information, or you can exit. Exit with the node deleted removes it. When the `DelNode` is selected, and the Key Verification UPF entry is selected, you are prompted to confirm the intent to clear the node. The default will be `"N"` for NO. Pressing `"Y"` for YES and `Enter` will perform the clear.

5.17.5 Global Output

The Peer Cop template for Global Output allows you to specify what data is to be sent from the configured controller to all local devices on the Modbus Plus network. The transfer takes place from the area of state RAM you select, to the Peer processors at the end of each logic scan. The sending Peer processor then transmits the data on each token-pass.

☞ **Note** Only one entry can be configured for global output.

☞ **Note** Configuring Global output in a controller (device) does not automatically mean that the destination devices will receive and / or process it. *The destination device must be configured for "Global Input" from the source device to enable the process.*

5.17.6 Global Input

The Peer Cop template for Global Input allows you to define what data to expect from other devices on the local Modbus Plus network. If the defined data is received, it is transferred to the specified state RAM area, from the Peer processor, at the start of each scan.

You may configure multiple entries for Global Input, with up to 8 entries for each device on the ***local*** network.

The source of Global Input data is a device on the local network. There can be between 1 and 64 device entries.

The "Length" and "Reference" fields define the destination for Global Input data and may be 0X, 1X, discretes or 3X, 4X registers. Discretes must be started on word boundaries (10001, 10017, 10033 etc.). The length is specified in words (16 descretes per word) and ***can be between 1 and 32 words long***.

The "Type" field is either assigned by Modsoft or selected from a list displayed when you press any key (with the cursor in the "Type " field).

The options are BIN or BCD

The Index field allows you to specify a partial entry into the defined length. This feature is useful when you do not need all the data and want to save state RAM space.

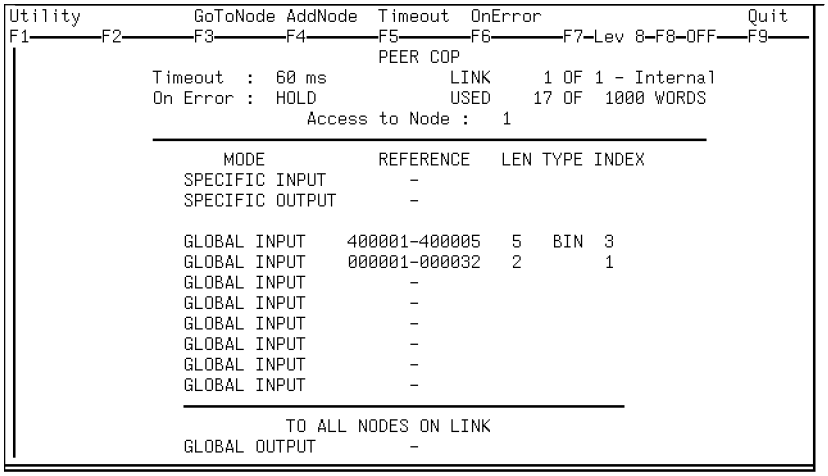


Figure 185 Global Input Screen With Sample Data

5.17.6.1 Function Summary

Using the previous example, you have a configuration extension where:

Five words are defined but the data accepted begins at the 3rd 4x reference of the global data from device 1.

Two 0X words are defined but only the second data word (bits 17-32) are accepted.

5.17.6.2 Momentum M1 Peer Cop

Peer Cop Programming in the Momentum M1 is identical to all other peer cop programming in *non-Quantum* PLC's. Quantum PLC's use a connotation of `IO Head` instead of `Link` used in non-quantum PLC. Modsoft will not prevent you from selecting link ID numbers other than 1.

5.18 S980 Address

This configuration Extension takes nine words and allows you to enter or modify the S980 interface network address. You should consult your network administrator for the proper data entry value. This allows the PLC to have a specific S980 MAC address attached to it. If for some reason the S980 was replaced with a new S980 (Different MAC Address) the PLC would be informed of this error. The menu allows you to Remove an address if you need to. If the configuration does not include the S980 block the error illustrated below is displayed.

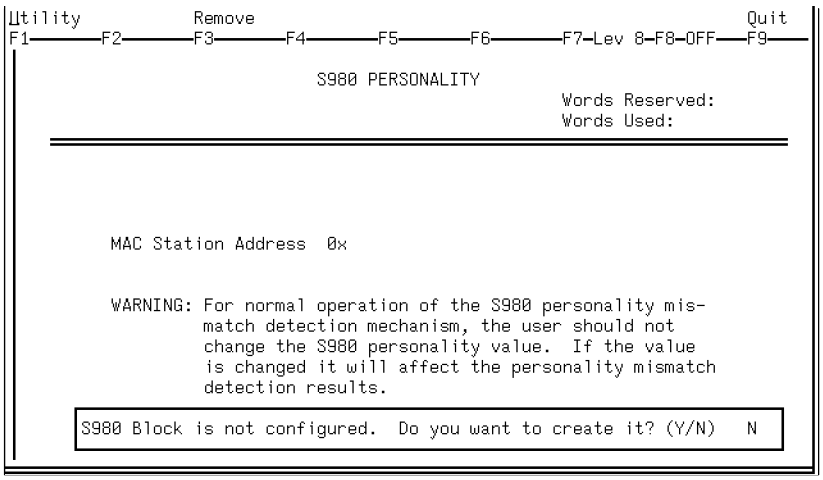


Figure 186 S980 Address Configuration Extension Screen

5.19 TSX QTM VME 486 Config Extension

5.19.1 VME 486 System Control Requirements

Initial `Config` selection and `OverView -> PLC Type and choosing the VME` allows you to select from an exec pack comprised of:

904
Q924
924V
424 0X

The 904 and 924V configure to 800 series I/O while the Q924 default to Quantum I/O.

VME 486 bus communications configuration is achieved largely by inclusion of a `configuration extension`. Your Ladder Logic program must allocate space for this configuration extension. The space allocation is fixed at 10 words.

There are 12 items that need to be viewed and set by the user.

- Slave Interrupt level (1-7) x
- Status I/D xxx
- Master Arbitration Type
  - Not System Controller
  - Primary Mode
  - Round Robin Mode
- Master Release Modes
  - Release on Request
  - Release When Done
  - Release On Clear
  - VMEBus Capture and Hold
- Master VME Bus Request Level (BR0-BR3)
- Interrupt 1 & IPL level enabled / disabled
- Interrupt 2 & IPL level enabled / disabled
- Interrupt 3 & IPL level enabled / disabled
- Interrupt 4 & IPL level enabled / disabled

- Interrupt 5 & IPL level enabled / disabled
- Interrupt 6 & IPL level enabled / disabled
- Interrupt 7 & IPL level enabled / disabled

The above data items are available under the configuration menu entry `Cfg Ext -> VME ConfigExt` for example:

☞ **Note** You must establish a `Cfg. Extension Size` and `VME controller type` selection before the VME `ConfigExt` selection item becomes available.

```
Hex      Dec      Bin      Goto                                        Quit
F1-------F2-------F3-------F4-------F5-------F6-------F7-Lev 8-F8-OFF--F9-----
                    TSX Quantum VME Config Extensions          Screen 1 / 1

      Slave Interrupt level: (or Delete Ext)  1

      Status ID (1-255)                       1          DEC

      Master Arbitration Type:                Not Sys Control

      Master Release Modes                    Rel. on Request

      Master VME Bus Request Level            BR 0

      Interrupt 1 & IPL Level:                Disabled
      Interrupt 2 & IPL Level:                Disabled
      Interrupt 3 & IPL Level:                Disabled
      Interrupt 4 & IPL Level:                Disabled
      Interrupt 5 & IPL Level:                Disabled
      Interrupt 6 & IPL Level:                Disabled
      Interrupt 7 & IPL Level:                Disabled
                      End of Quantum VME Cfg. Extension
```

Figure 187 TSX VME 486 Configuration Extension

Place the cursor on the desired data field (use arrow keys) and press the return key to either toggle the entry or display the list to select the data entry form.

5.20 Quantum Hot Standby

Access to this Config Ext. feature is not dependent on the existence of the Quantum Hot Standby loadable in the ***Modsoft\Loadables directory*** i.e. you do not have to install the loadable first. Modsoft warns you if the config ext is present and the Controller Hot Standby (CHS) loadable is not present when exiting from the config screen. When downloading to the controller Modsoft does not include a check for existence of the CHS loadable if the config ext is present.

The config extension for Quantum Hot Standby replaces having to program the Hot Standby function block in network 1 of Segment 1. The data in the config ext area is as follows:

Config Extension Length	word 1
Control/Status word	word 2
Block type = 5	word 3
Command Register	word 4
Initial Cmd Reg value	word 5
Non-transfer Area start	word 6
Non-transfer Area length	word 7
State Ram Xfer Control	word 8
Option block 1	4 words
Option block 2	4 words
Checksum	word 17

Word 8 has the following layout:

Bit 0: set: (default)User defined state ram spec (data in option block 1).
not set: Standard. 12k state ram hsby (no data in option block1).

Bit 1: set: Additional state ram transfer on multiple scans.
not set: No state ram transfer beyond first scan.

Bit 2: set: override bits 0,1 defaults and transfer all state ram.
not set: No override bits 0,1 default action.

Bit 3: unused

Bit 4: set: If 3x size >= 4x size of an additional state RAM transfer.

Bit 5: set: if 0x size >= 1x size of additional state RAM transfer.

Bit 6: set: if 3x size >= 4x size of state RAM transfer.
bit 7: set: if 0x size >= 1x size of state RAM transfer.

Bits 8-15: Extra transfer time: Maximum number of scans to transfer additional state ram (see bit 1)

Note: Bits 4-7 are designed to prevent accidental swapping of discrete and register length without being detected by the checksum.

Screen Layout...

The page 1 screen for hot standby is identical to the zoom screen for the HSBY function block. Page 2 departs from the HSBY block zoom.

Figure 188 Quantum Hot Standby Configuration Extension Screen 1

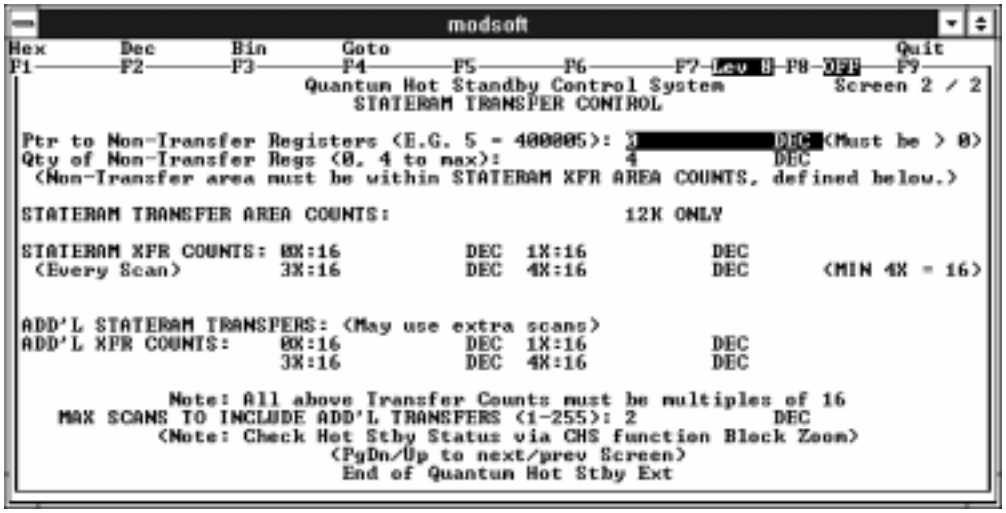


Figure 189 Quantum Hot Standby Configuration Extension Screen 2

This Config Ext editor is implemented using the Zoom screen compiler, and special supportive code, much like the VME control config extension. This technique allows relatively fast implementation but does imply certain limitations on the user interface, especially the validation of data entry. Entry of the Control and Non-Transfer Registers however, must be done as a pointer to the register and not as the full register name; i.e. 5 is the pointer to 400005, (or 40005). The Zoom editor does not handle reference data types. Register numbers entered are not validated against the max number of registers configured.

5.21 TCP/IP

5.21.1 TCP/IP Protocol Requirements

The Transmission Control Protocol/Internet Protocol (TCP/IP) module (NOE module) is configured using the configuration extension area for operating parameter entry.

Parameter	Size (Bytes)	Description
ip_addr	4	32 bit internal address
enet_frame	1	Ethernet Framing Type
qbpslot_id	1	Quantum slot ID
unassigned	26	
TOTAL	32	

The Configuration Extension screens to view and input this data require four separate entry fields for the internet address, four more to define the gateway path and one each for the slot and the framing type. The framing option is between Ethernet II or IEEE 802.3

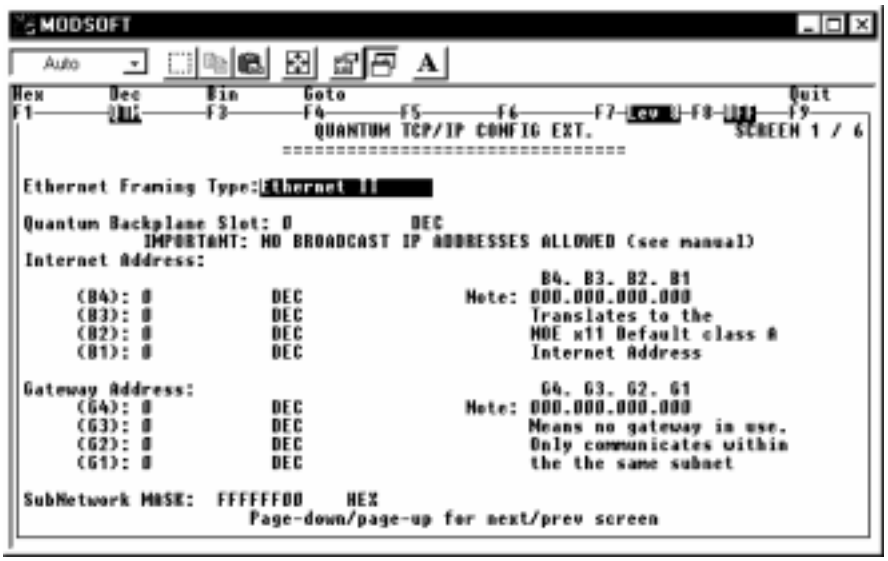


Figure 190 TCP/IP Interface

Editing data within this configuration extension is accomplished using the Modsoft Zoom screen compiler and special support code.

5.21.1.1 Multiple TCP/IP Slots

In the 486 Quantum there can be six different TCP/IP modules in operation, therefore their configuration is handled as six separate pages of zoom screen. Data entered in the zoom screen is processed when Modsoft exits that operation and it is arranged in one contiguous extension area. A slot number of zero is a signal that there is no valid data on that page. If it is ignored and is not in the configuration extension.

5.21.1.2 MSTR Block Help and Zoom

In the TCP/IP implementation, the MSTR block screens are modified to include information concerning the TCP/IP on the second page. A third page is included for the Quantum SY/MAX Ethernet option modules.

Note In pure DOS environments, and in windows 3.x, a DOS TCP/IP stack (TSR) must be loaded as well as a network module (netmod) that corresponds to the stack-provider. For Windows 95, Windows NT and OS-2 the netmod only is used to translate TSR calls to winsock.dll calls. In either case, the `netmod` must be specified in the VSLDOS.INI file. Refer to Appendix I for software support details.

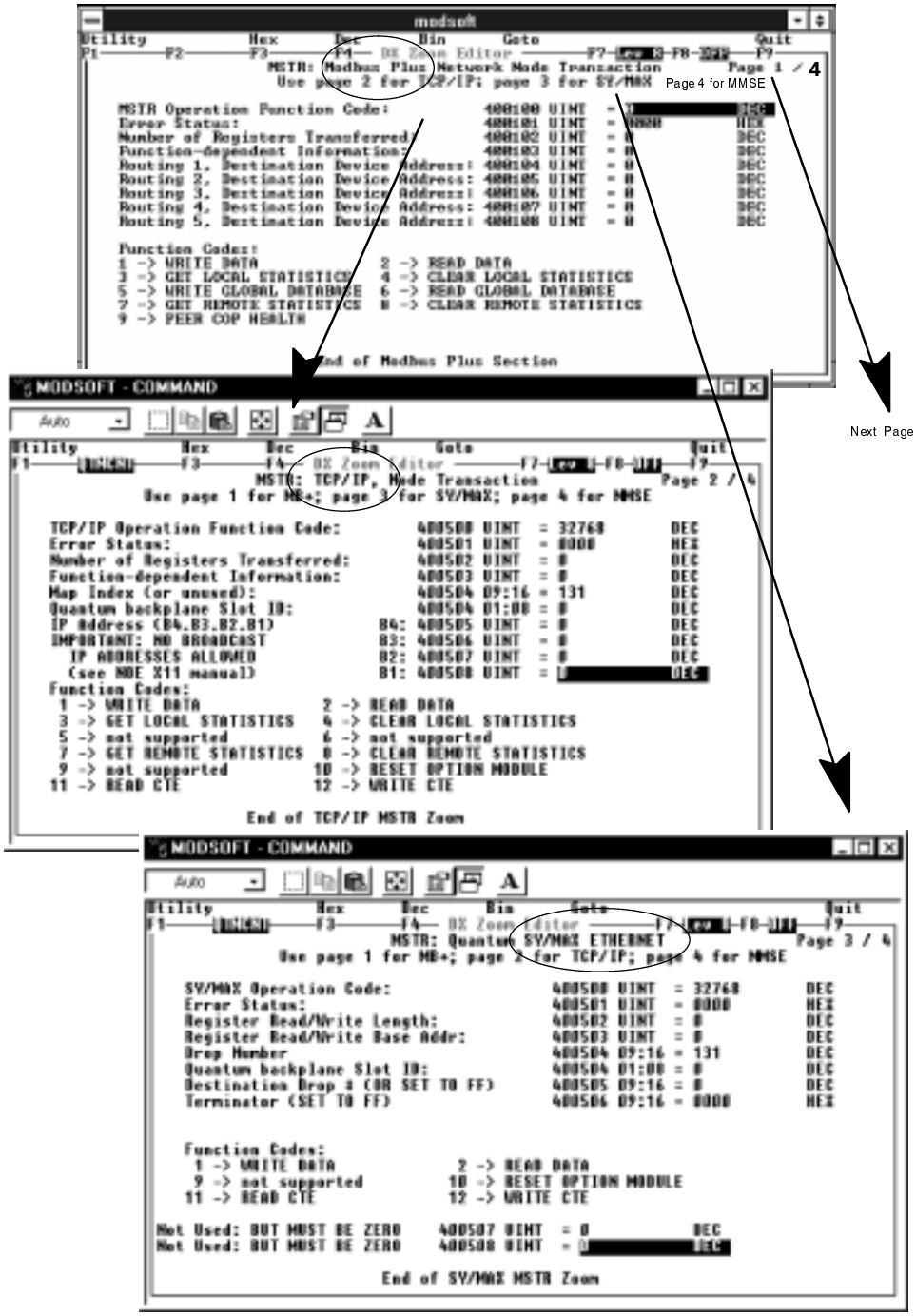


Figure 191 MSTR Zoom screens for Modbus Plus, TCP/IP abd SY/MAX Ether net Setup

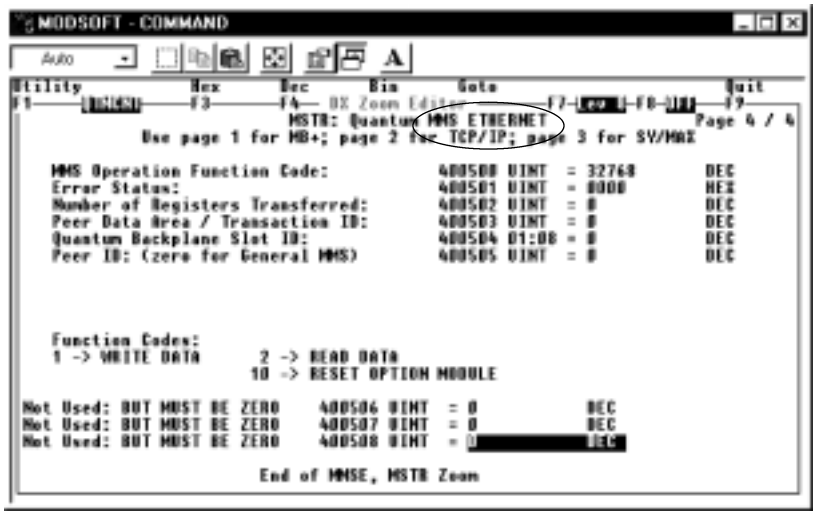


Figure 192 The MMSE Zoom Screen For MSTR

5.22 SY/MAX Ethernet Setup

This is the last function available from the Configuration Extension menu. It is only available if you have defined the SY/MAX S908 inclusion in the UPF editor "`General` → `Miscellaneous`" Menu.

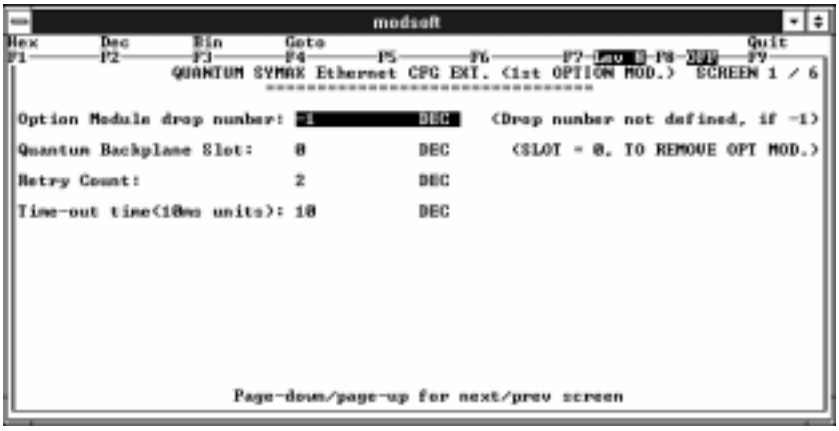


Figure 193 SY\MAXEnet Cfg. Ext

Six identical CFG EXT screens are offered, one for each possible option module. You must supply the Quantum Module Drop and Slot assignment numbers. In addition, the communications retry count must be set, (if different from the default), and the"Communications timeout" value can be adjusted from the default of 10ms.

5.23 Quantum MMS Ethernet

This Configuration Extension is provided to configure the Network Option Ethernet (NOE) for Manufacturing Message Specification communications protocol modules in an Ethernet environment. When you select this extension you are provided with data zoom screens that allow you to set the routing addresses necessary to traverse the communications network. These parameters include:

- The option module slot number in the backplane
- Power ON
- Reset conditions
- Network Service Access Points (NSAP)
- X500 NSAP strings with Server option
- Local PEER Identification

☞ **Note** The Quantum 186 Controllers support a maximum of 2 Option boards and the Quantum 486 controllers support up to 6 Option boards in the local backplane.

There are two parameter entry screens required to support one option board, therefore there are 12 screens in this zoom (2 x 6). The following two screens are shown as examples:

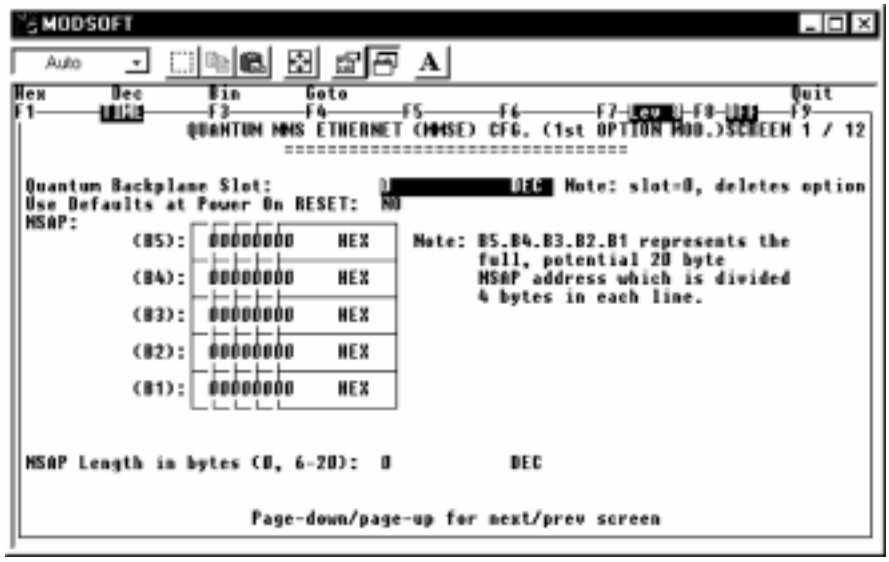


Figure 194 First MMS Option Screen One

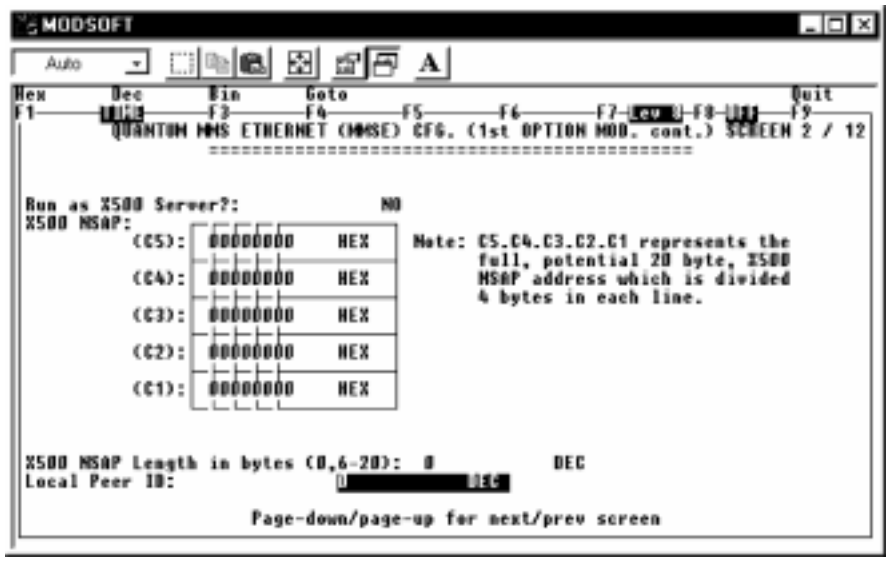


Figure 195 First MMS Option Screen Two

5.24 Online/Combined Configuration

When the Online configuration menu component is activated, the actual controller configuration is transferred to the Panel and shown on the screen. The configuration menu in Online mode presents the PlcOps "I/O Map", "Ports", "Segments", "Loadable", "Configuration Extension" and "Quit". All changes within the configuration have to be done with the controller in Stop mode. A parallel update of the database is provided if you are in Combined Mode.

5.24.1 Configuration Extension

The "Cfg Ext" menu items "ReadExt" and "WrtExt" are enabled in the online and Combined modes. You can Read the current Configuration Extension, Edit it and Write it back to the Controller, *as long as the amount of memory declared for Configuration Extensions is not exceeded*

Caution Writing the Configuration Extension ONLINE causes the PLC to disappear from the MB+ Network while the command is in progress. Other devices communication with this PLC will receive errors indicating that this node is unavailable.

5.24.1.1 I/O Map

The status of I/O modules is displayed in the configuration editor when the menu item "I/O Map" has been selected. If the controller is stopped, the same window (as in Offline mode) is displayed, with the last column (description) used differently. The information is taken out of the controller's configuration table. The right side of the screen shows the modules physically put into their corresponding slots, so you can compare the configured modules with those actually available. If the controller is running, a reduced display only shows whether the I/O module health is OK, as indicated by an * beside the Module ID (the full I/O module status is accessible only if the controller is stopped). The values in the I/O Map can be changed.

Caution On a Compact 984 adding modules that require a loadable driver, the controller will not start because the loadable cannot be added on line. You must go offline to add the new driver to the controller configuration.

5.24.1.2 Ports

The parameters of the Modbus Port can be modified when " Ports" is selected. The same off line style window is opened. The changes are written to the controller after the modification for the port has been done. Before writing to the controller, a warning is issued that the modi-

fied port parameters may abort the communication between programming panel and controller. You must confirm this warning explicitly.

You can view Loadables but NOT change them in Online mode (PlcOps are not allowed in this area).

5.24.1.3 Segment Scheduler

The menu in `Online/Combined` mode provides the standard utility. `PlcOps` are not allowed in this area of Modsoft.

Figure 196 Segment Scheduler While Online

Chapter 6
Ladder Logic Programming

- Ladder diagrams allow you to design control networks for automated operation of complex hardware and time dependent operations.

6.1 Ladder Editor

6.1.1 Ladder Edit Initialization

The Editor selection available from Offline, Online or Combined programming modes, provides you with the functions required to implement control logic using Ladder diagramming techniques.

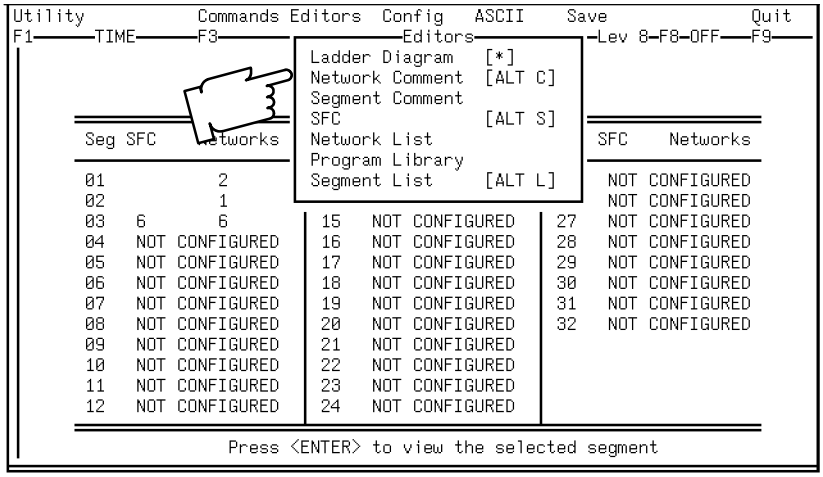


Figure 197 Entering The Ladder Diagram Editor

In the above illustration, the “Time” example is used. When you place the cursor in the segment 1 field, and press return, a network is displayed. You can use the **<PgDn>/<PgUp>** keys to display multiple networks.

Other Ladder diagrams are entered by moving the cursor to the segment of interest and then by pressing the ↵ key.

Each segment of Ladder Logic has the number of networks displayed. In the case of the example used in this document, there is one network in the first segment. In addition to entering the Ladder Logic and associated editors, you can also alter the segment selected or interact with the Network Documentation Commentary by selecting the features from the Ladder main Menu or SFC Menu. The Ladder Editor Menu comprises the following functions:

- Offline
  Utility Elements Edit Go/Srch Network Refs Tools Quit

- Online and Combined
  Utility PlcOps ElementsEdit Go/Srch Network Refs Tools Quit

The Ladder Diagram Editor (LDE) provides the ladder diagram objects and interconnects that make up the networks. The ladder diagram editor is used to manipulate individual networks in graphic form. In order to build up and modify a ladder diagram, two groups of commands can be distinguished:

- One group provides the elements of the ladder diagram language
- The other provides the functions to manipulate the elements and provides general editor functions.

For a detailed definition of attributes such as Reference Symbols and Network Comments, you can select other editors that handle these tasks. Your view of a network consists of a row/column matrix, which contains a graphic representation of operands and operations (called a Node) that may be connected by horizontal and vertical links. In the first line of the ladder diagram window, the network number and an optional network title are displayed, The bottom line displays the symbol of an operand, including the descriptor. This information is changed whenever the cursor moves from one node to another.

6.1.2 Ladder Display

When you enter the Ladder Logic Editor by cursor selection from the Segment display, the shorthand for the Ladder Editor is < * >. The screen of the ladder diagram editor is separated into several areas. In the top line, a menu bar for the command groups is displayed.

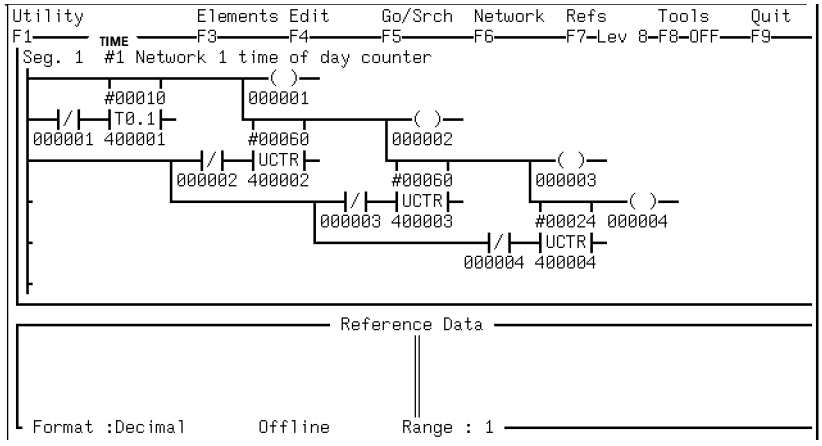


Figure 198 Initial Screen Of Offline Ladder Diagram Menu Selection

The rest of the screen is the work area. An ID line is displayed to remind you of the Segment displayed, the network in the segment and the network number as it related to the total of all networks in this program. The layout depends on the chosen display mode: reference or symbolic. When the reference display mode is selected, the window is divided into node elements of 7 characters wide and 2 rows high. The seventh character column is used for vertical shorts. This allows display of a network of 11 x 7 nodes. The cursor is represented in reverse video mode, the same size as one node element. A horizontal and vertical wrap is provided. In this mode, references can be input, and the lower part of the screen is a window to the reference data editor. Thus, when running online, you get access to information related to a certain network as well as to the state of some chosen references. (Refer to the Reference Data Editor Chapter.)

When you switch to symbolic display mode, the window is enlarged vertically and horizontally. The screen is completely covered by the ladder diagram editor and the height of a node element is then 3 rows (one additional line for the assigned symbol) and is the width of 11 characters (10 for the symbolic name and one for vertical shorts). Thus, for 11 column networks, a horizontal scrolling of the screen within the symbolic display mode is provided. This mode allows you both reference and symbolic operand input. The wide screen can be activated by the shorthand toggle <**Alt W**>.

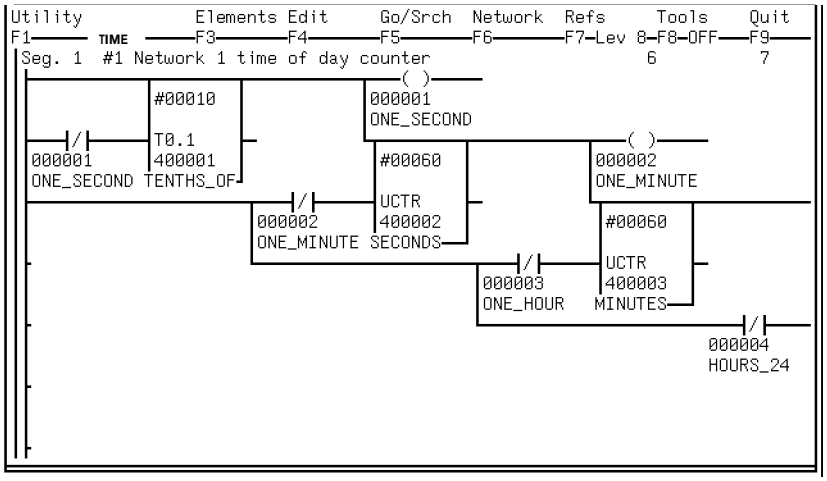


Figure 199 Example Of Symbol Mode Screen Expansion

While in the Ladder editor, if the Reference Data Editor is selected the RDE window becomes active underneath the LDE window. The menu bar changes colors to signal this, and a cursor appears in the RDE window. Only the first 5 rows of the first RDE page can be viewed or edited. A total of 10 references can be seen, although all 88 references are still stored underneath.

If the Symbol is greater than 10 characters a - and > are used to signify that condition. Using the 400001 reference above for example, if the symbol were TENTHS_OF_A_SECOND the display would be: TENTHS_O->. (Additionally if there is no symbol present the first 9 characters of the Descriptor are placed in the symbol field).

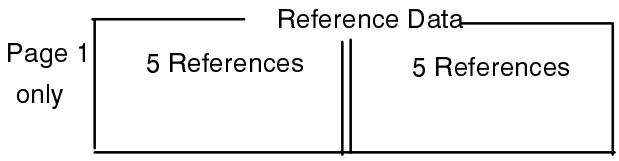


Figure 200 Reference Data Area

6.2 Ladder Utility

The Utility functions within the Programming types are distributed slightly different than offered at the main menu.

Utility Function Comparison

Offline	Online and Combined
Key Help	Key Help
Program Information	PLC & I/O Status
Reference Data	Program Information
Symbol Table	Reference Data
Dos	Symbol Table
	Dos

6.3 Ladder Diagram Elements

Ladder diagram elements are schematic-like electrical elements, including connections between them. The objects are depicted graphically and can be chosen by the "Element" pulldown menu, which shows the different groups of operators and an associated selection list. The selected object is displayed at the current cursor position. After the element is displayed, you can insert the appropriate reference or parameter. A full description of the application of Elements is found in the Modicon Ladder Logic Block Library 840 USE 101 00.

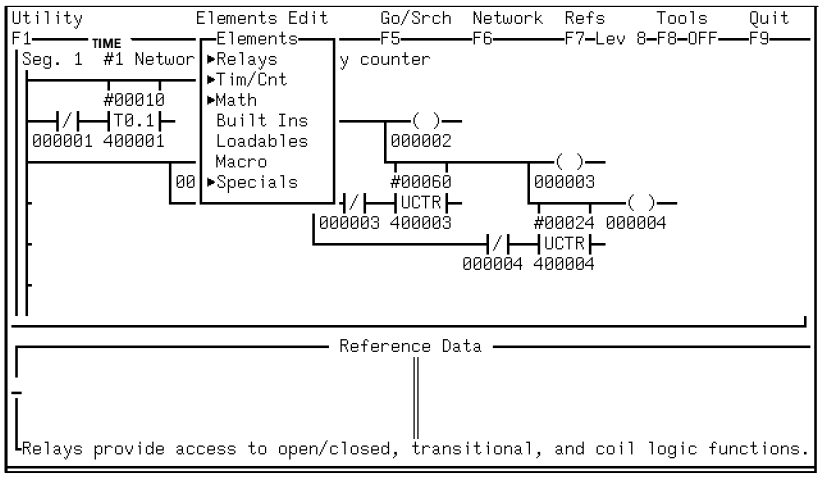


Figure 201 Ladder Diagram Element Pulldown Menu

☞ **Note** To view a specific help screen for any element selection, the cursor must be highlighting the item in the pulldown menu; then type <**Alt H**>. Help is also displayed by placing the cursor on the currently displayed logic element and typing <**Alt H**>

The identifiers are type checked for whether the input operand matches the actual ranges defined in the configuration. The second line within a node keeps the absolute reference, while the third line shows an optionally assigned symbol (in symbolic mode). When editing in symbolic mode, the symbol line is accessed first, before you are asked for an reference operand. When entering reference operands the following apply:

- The allowed ranges (according to the actual configuration) are shown in the bottom line of the ladder diagram editor.

- If you do not want to define the operand yet, you may confirm the “?????” preset by pressing ↵.
- Operand inputs shorter than five characters are filled with leading zeros (0XXXX reference).
- Although constants are internally kept with a leading “#”, you may type in constants without this constant indication except when using Quantum, 785L or 785E where any value >9999 must be preceded with the # sign.
- Symbols can be entered only in symbolic display mode (Screen Size).

If you specify a symbol that is not already defined in the symbol table, the cursor is automatically moved up to the middle line of the node, where you must specify the reference address. This entry is then inserted into the Symbol Table.

6.3.1 Relays

This submenu provides the selection of the schematic symbols for various relay contact types.

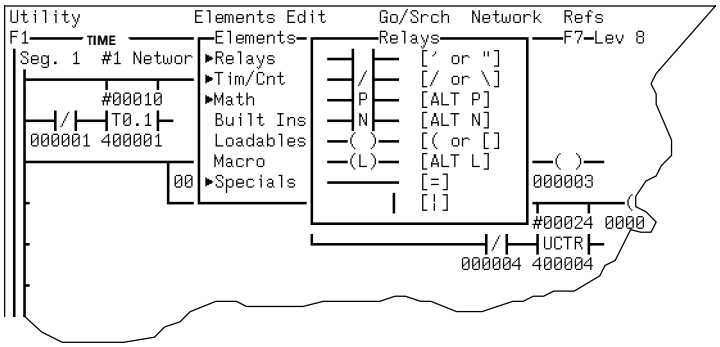


Figure 202 Ladder Relay Selection

When inserting coils, the Coil Used Table is checked and if a coil is already assigned any attempt to add another one is rejected.

6.3.1.1 Timers and Counters

Timers and counters are available for assignment with the following parameters:

- Timers

  T1.0 = One second
  T0.1 = One tenth of a second
  T.01 = One hundredth of a second

- Counters

  DCtr = Down counter
  UCtr = Up Counter

6.3.1.2 Math

The math function blocks for ADD, SUB, MUL and DIV are selectable from this pulldown.

6.3.1.3 Built Ins

These are a specific group of logic block elements supplied with Modsoft. ***The number and type of built-in functions are dependent upon the PLC selected in the Configuration.*** Both Tracking Help and <Alt H> specific help are available. If you want to select the element directly just type it in the cursor areas of the logic diagram. For example selecting an “DIOH” <return>, in the block illustrated, would produce the Dx function block at that location.

Some builtins can be viewed on the screen, using the DX Zoom function, to observe status of the logic block or in some cases to establish block initialization.

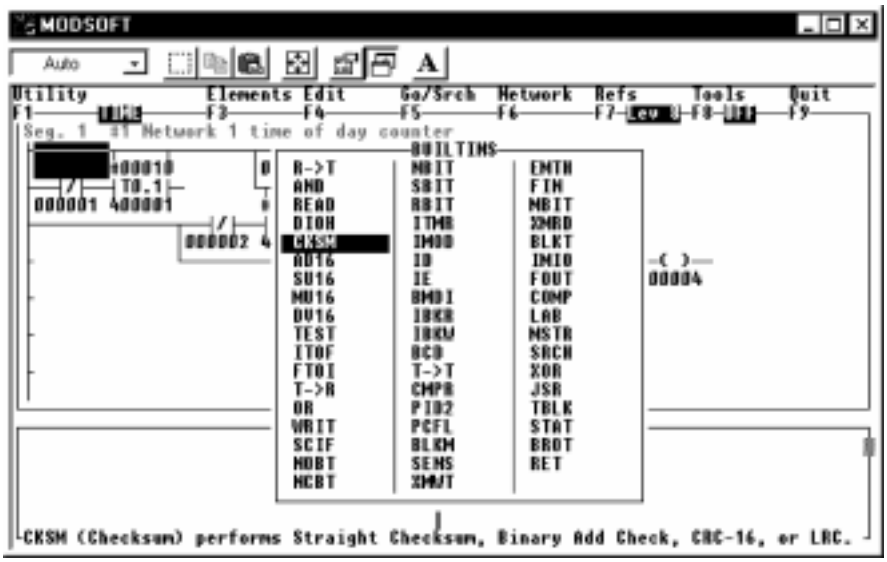


Figure 203 Built In Block Selection List (Quantum Selected)

Figure 204 Builtins if M1 Selected

The procedure you use to select the Dx Zoom feature is:

1) Locate the cursor on the lower node of the block

2) Press the Tab key to reposition the cursor on the menu select line where the **Refs** pulldown allows you to select `Dx Zoom`.

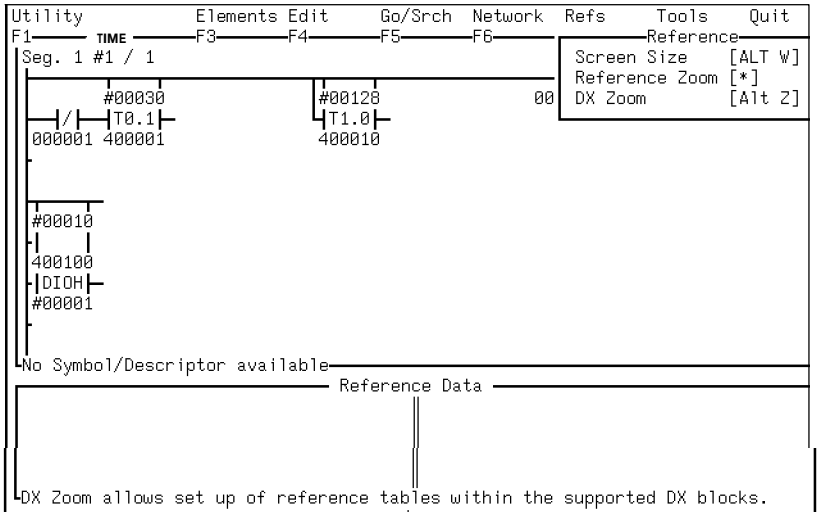


Figure 205 Setting up a Dx Zoom for a Builtin Function

The results for the above example is illustrated in the next illustration.

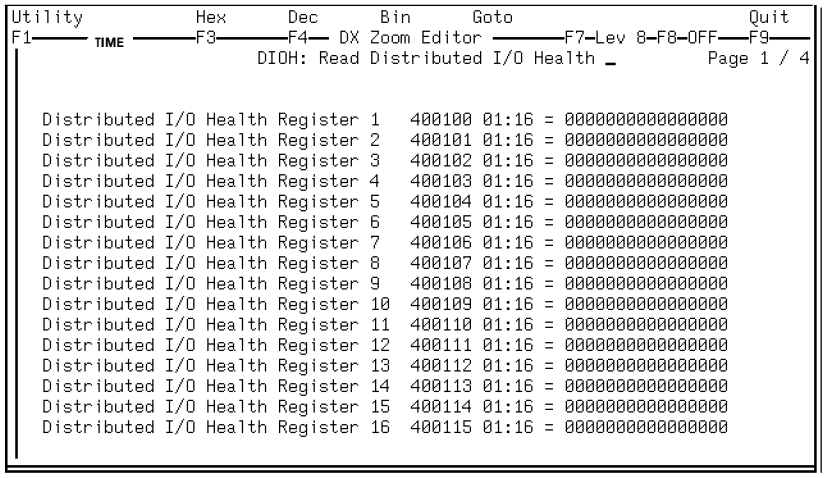


Figure 206 A Dx Zoom Example Screen

- EMTH

All 38 of the Extended Math functions (for 685E, 785E and Quantum) as a selection listing are provided. You can activate the list by placing the DX block cursor on the bottom node and

pressing the < **?** > or return keys. There is a help screen available for each selection. Put the selection cursor on the desired item and press the help keys <**Alt H**>.

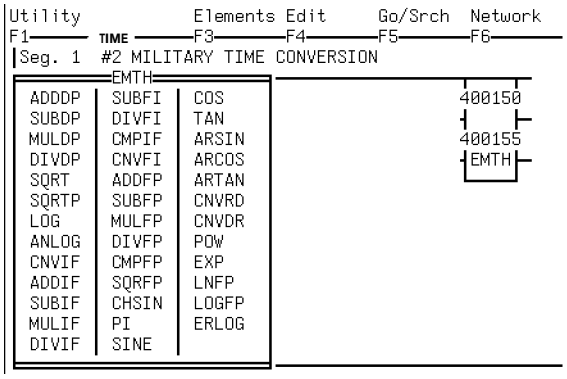


Figure 207 EMTH DX Function Block and Select List

- PCFL

The Process Control Functions (for specific Modicon controllers) are provided as a selection listing. You can activate the list by placing the DX block cursor on the top node and pressing the < ? > or return keys. There is a help screen available for each selection. Put the selection cursor on the desired item and press the help keys <**Alt H**>.

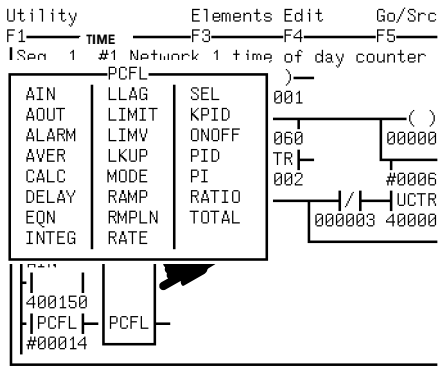


Figure 208 PCFL DX Function Block with Library Selection List

6.3.1.4 Loadables

Loadables are specific design elements that are provided or procured separately from Modsoft but can be included in your program as programming elements. These elements have to

be loaded via your configuration so that the Ladder Editor can access them. The Help assists are derived from the Loadable.DAT file.

Quantum Loadables in Ver. 2.4

When the Quantum PLC is configured, the following additional loadables can be loaded

CTE, CON, ESI, NOL 1x3x

These elements are complete with help screens and zooms. The block data is included for the new Quantum controllers only. The inclusion of these elements results in "reduction" of the Node ID numbers to be used by loadables. You may have to modify the OP code selection to accommodate a given loadable. For a description of available Loadables refer to The Modicon Ladder Logic Block Library 840-USE-101-00

Quantum PLC Executive ID

The new quantum controllers have new exec ID's. Some existing programs will not operate correctly if they have loadables with the same node ID. Consequently it is necessary for Modsoft to test the opcodes of loadables against the codes in the opcodes.sys file during the preparations for download to the controller. You then have to modify the loadables list to select different opcodes to allow download.

6.3.1.5 MACROS

Macros are specific functional logic sequences that you design and implement as a programming element using this element selection. Macro development is discussed in Chapter 11.

☞ **Note** The SFC flag must be turned to **ON** for Macro Programmed elements to be used in a Ladder Logic program.

6.3.1.6 Special Elements

The SKIP operator skips to a given network number within the step (or Segment), in the forward direction. For example, SKIP 3 skips to network #3 in the current segment step. The SKPC operator inserts a SKP node with a constant parameter (Skip Constant). The final skip type is the SKPR, which inserts a SKP node with a 3XXXX or 4XXXX register reference.

The operand of the "SKIP" object specifies the reference target network number of a jump and not an offset. A jump out of the segment is not possible. The correct SKPC offset is then calculated when the program is converted before downloading.

The “SKIP”, “TC”, and “RSTF” functions are not available in ONLINE Mode.

☞ **Note** The operator “SKP” is not available within the ladder diagram editor. Instead of this, operators “SKIP”, SKPC and SKPR are provided.

6.3.1.7 Ladder Diagram Shorthand

The following shorthand is available for editing ladder diagram elements:

‘ or ”	for a normally open contact
(or [	for a coil
\ or /	for normally closed contacts.
=	for a horizontal short
|	for a vertical short
Alt N	for negative transition contact
Alt P	for positive transition contact
+	Use one more reference than previously entered
-	Use one less
Space	Use the same reference
,	Go to symbol table and use reference of last line visited in Symbol table.
PgUp	Previous Network
PgDn	Next Network
Home	Row 1 Column 1
End	Row 7 Column 11
Ctrl-Home	First Network of Segment
Ctrl-End	Last Network of Segment
Ctrl-PgUp	Previous Segment
Ctrl-PgDn	Next Segment

Type the name of DX function block desired at cursor location on logic screen.
Control PgUp Toggles between segments
Control PgDn Toggles between segments

The following is an example of the “Time” ladder diagram.

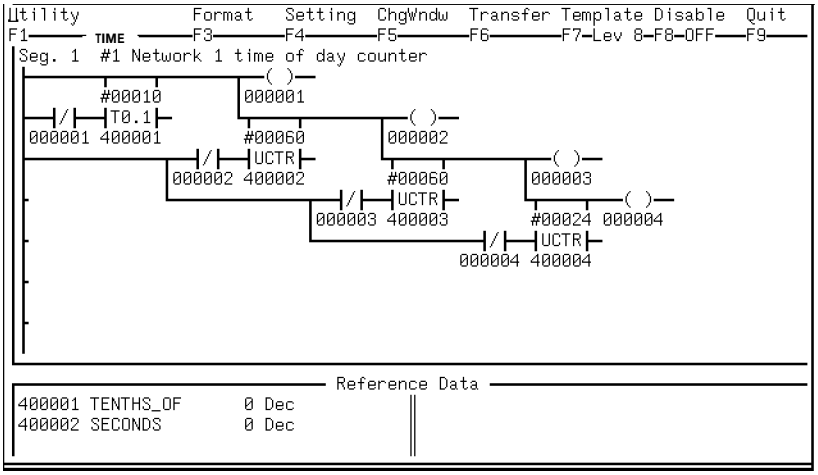


Figure 209 "Time" Example Ladder Diagram

☞ **Note** A coil can be inserted only at the rightmost position of all objects entered on one editor line. If there is a coil inserted, no object can be inserted to the right of the coil. Column 11 is reserved for coils only. If a duplicate coil is entered it will be detected during program conversion prior to downloading to the controller.

6.4 Ladder Logic Specific Edit Pulldown

The Editpulldown provides the tools for the physical construction of your Logic.

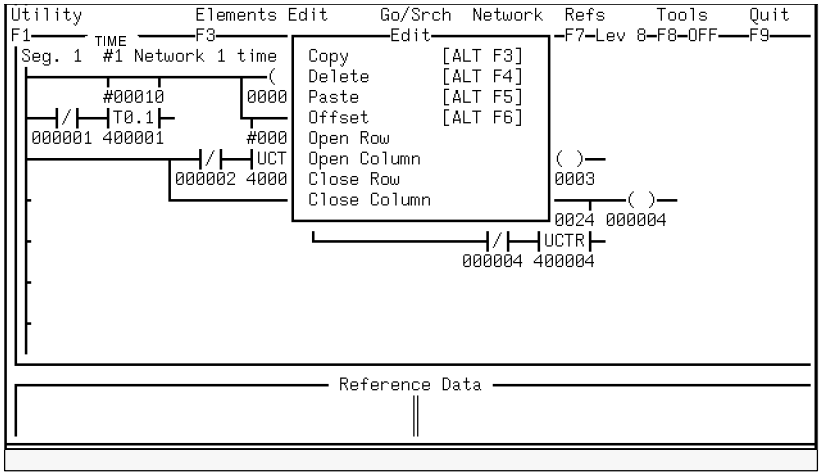


Figure 210 Ladder EDIT Selection

6.4.1 Copy

The marked data (Cursor selected) is copied into the paste buffer. *There is only one paste buffer for the ladder diagram editor, and it is used by both the Copy and Delete commands.* The shorthand keys available for this and other commands are shown. The shorthand key is <**Alt F3**>.

6.4.1.1 Delete

Using this command, you can open a Delete frame, the size of which can be changed by use of the cursor keys. The cursor position when you select the function defines one edge of the frame, and the current cursor position is diagonally at the other corner of the Delete frame (displayed in a special Delete color). This specified area is deleted from the work window and inserted into the paste buffer when <**Enter**> is pressed. Ladder diagram objects that occupy more than one node element, such as function boxes, are deleted only if the whole box is inside the Delete frame. Otherwise, the Delete operation is cancelled, and an error message is displayed. If <**Esc**> is pressed, the Delete function is aborted, and everything stays unchanged. The shorthand key is <**Alt F4**>.

☞ **Note** You can only `block delete` nodes when in offline mode.

6.4.1.2 Paste

This function inserts (at the current cursor position) the contents of the paste buffer which was most recently filled by the Delete or Copy command. This function can be invoked only if there is enough free space to hold the contents of the buffer. The Delete/Copy/Paste mechanism functions across different networks, so you can delete a part of one network and paste it in another one. In addition, this Paste network may come from a program in a library. The shorthand key is <**Alt F5**>.

This operation can take a long time to perform if a large number of networks are selected at the same time. Furthermore, it is possible that in doing a large number of networks at once, the database operations might fragment the memory pool, and eventually run out of memory due to not being able to find a piece large enough for another allocation. To smooth out this operation do ten or so networks at a time, and then exit the editor, followed by saving the program. In a severe case it may be necessary to also exit from modsoft after saving the program.

6.4.2 Offset

Using this function you can shift some operands within the underlying networks to another address. First, you are prompted for a range of references which should be shifted and for an offset value for this modification. Then, if the transformation is valid (according to the actual configuration), you must define the ladder diagram nodes where the modification should take place. This area is specified by marking the cursor from the upper left of the ladder logic to the lower right ladder diagram bound. The number of replacements is reported when “Offset” has been executed. The shorthand key is <**Alt F6**>.

This example is established using the “Time” example in segment 1.

Note To offset DX functions with coil references you must use an offset That is a coil boundary (multiple of 16+1) i.e., 17, 33, 1793, 2801 etc.

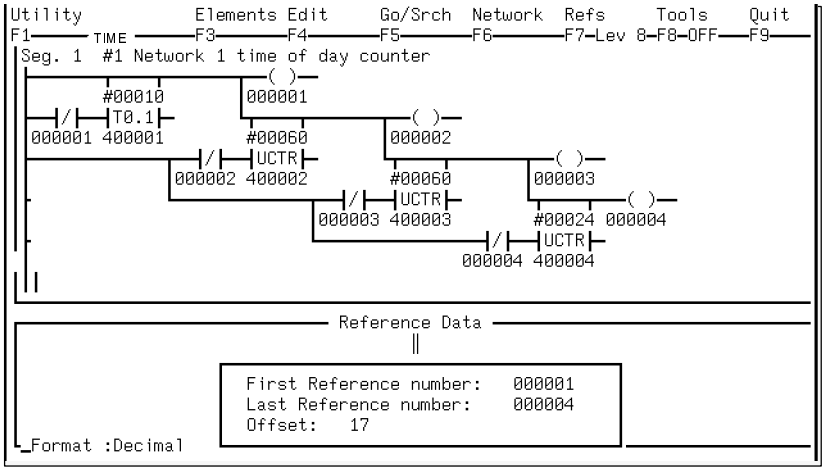


Figure 211 Offset Display Prompt

If the configured references are valid you then mark the logic where the modification should take place. This selection is cursor specified which means you move the cursor over the logic that makes the selection as noted by the network nodes being highlighted. The final selection is done by pressing the < ↵ > key. The number of replacements is reported when “Offset” has been executed. This function can also be called by pressing the <**Alt F6**> keys. The next figure illustrates your selection and the results.

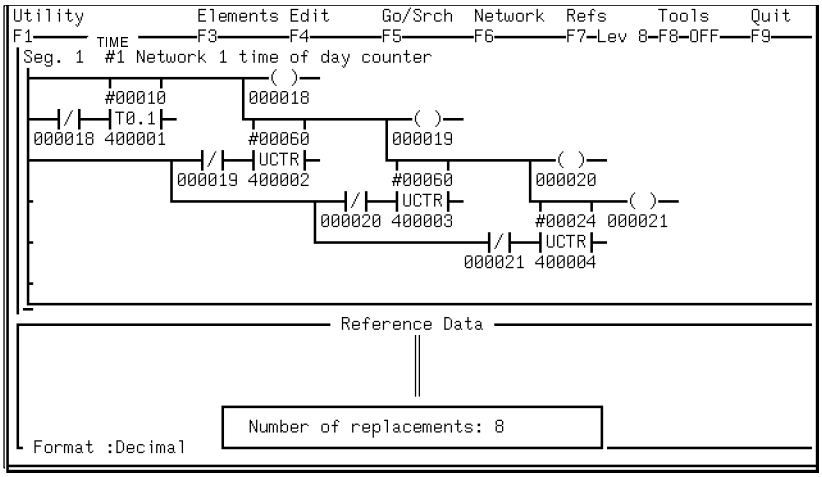


Figure 212 Offset Executed Result

6.4.2.1 Open Row

A new row is opened at the current cursor position. This command is executed only if there is enough free space (i.e., the last row is empty). The rest of the network is shifted down accordingly. Function boxes and other objects with a height of more than one node are not split by this command.

6.4.2.2 Open Column

If the rightmost node column is free, the rest of the network is shifted right, and an empty column is opened at the current cursor position.

6.4.2.3 Close Row

If the node row on which the cursor is positioned is empty, all node elements below are shifted up one row, and an empty bottom row remains.

6.4.2.4 Close Column

If the node column on which the cursor is positioned is empty, all node elements to the right are shifted left one column and an empty right column remains.

6.5 Go/Srch

The “Go/Srch” pulldown menu contains commands related to position your programming operation at a specific point in the constructed network. The menu also shows you what shorthand keys are available.

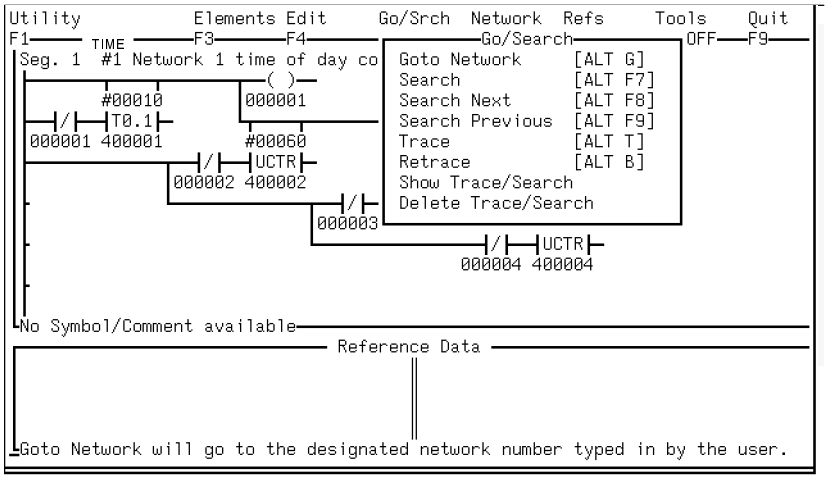


Figure 213 Go/Srch Selection Pulldown Menu

Every Trace, Search or GoTo places the current network location (before moving) into the Trace History. This Trace history acts as a LIFO stack. Every trace, search or goto adds to the stack of the history data. Doing a Retrace takes the last location from the Trace History and moves the cursor to that location. As a result, you can do as many retraces as there are history points in the buffer.

6.5.1 Go To Network

This command serves as a method to access networks that are not adjacent to the current network. The found network is displayed in the work window.

The special keys <**PgUp**> and <**PgDn**> bring up the previous and next network into the work window of the ladder diagram editor, if there is any, and <**Ctrl + PgUP**> or **PgDn** can be used to cross segment boundaries. The shorthand key is <**Alt G**>.

The <**Alt G**> can execute the Goto in different ways depending on program conditions. If SFC mode is “On” you are given a display prompt for the parameters:

Network Element Segment Network title

If you do an online Quick or are running with SFC OFF you are only allowed the GoTo parameters:

Network NetworkTitle

Use the return key to move from field to field. Pg Up/Dn accepts the changes

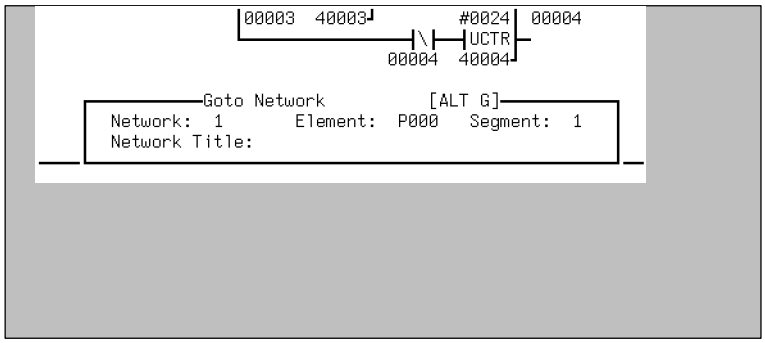


Figure 214 "GoTo Network" CommandWindow

Expert If SFC is On the information in the GoTo window is as illustrated. With SFC Off Element and Segment are not displayed and the Network # is changed to a Program offset not a segment offset.

6.5.1.1 Search

The search command in the Reference pulldown menu allows you to enter a search pattern for constants, references or their symbols.

Note: For logic containing Equation networks refer to the next chapter for special search considerations.

The search starts at Network #1 in the segment and proceeds downwards to the end of the program. If the search is successful, the cursor is positioned at the corresponding element. If the pattern is found (the pattern is not case sensitive) in another network, this network is displayed on the screen as it was shown.

To search for an Element, such as an open contact (-| |-) you can select `Go/Srch` and the subfunction `Search`. Then:

1) Fill in the `Search For` information in the Search prompt dialog box.

2) Put the cursor on the `Node Type` field in the Search prompt dialog box, and press F1 then F1 again to select the open contact. OR:

3) Put the cursor on the `Node Type` field and press the Quick Key for the element you want to search for. When you select this function if you do not want to specify a four character “Node” type DX (MSTR etc.) just return after entering the reference type.

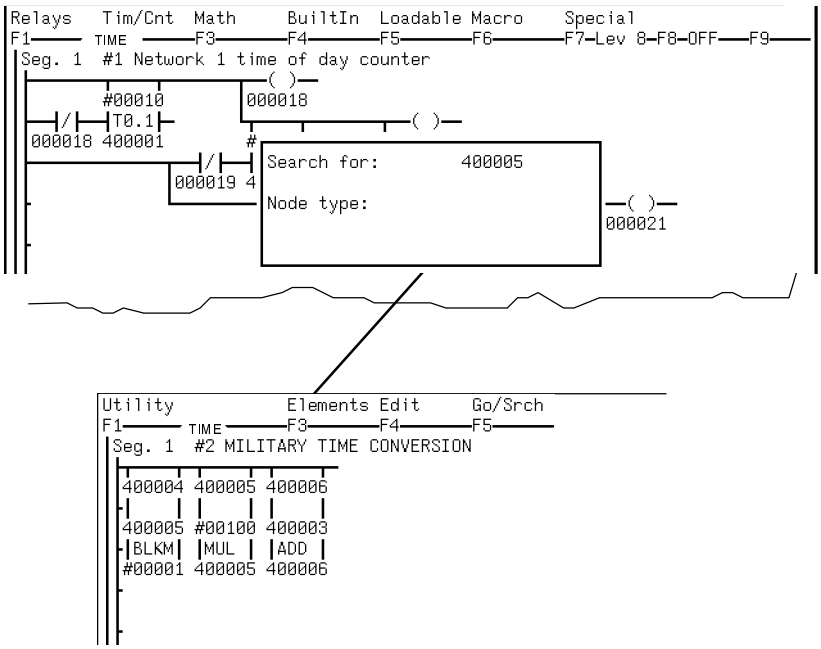


Figure 215 Illustration of Reference Search

6.5.1.2 Search Next

After a successful search, the search pattern can be searched again without the necessity of reentering the search pattern. If no valid search pattern is available, an error message is issued. The shorthand key is <**Alt F8**>.

The last reference number searched is remembered and available as a default. Subsequent `Search` selection reverts to the default string. If you follow the search with a trace the search value is retained.

☞ **Note** If Offline Search Next/Previous searches from the cursor location. If Online the activity is from the place where the last search found the item.

6.5.1.3 Trace

This function is used to determine the relationship and location of “coil / relay” references. If the cursor is positioned at a location with a 0XXXX reference, the associated coil can be found by the trace function.

In the example the cursor was first placed on the 00001 contact. The coil location found is reported, and the cursor is positioned on this coil node. If no coil is found, an error message is issued. The shorthand keys are <**Alt T**>.

6.5.1.4 Retrace

After a previous trace function, the original network is restored by the retrace function. If using the example below, the cursor would be reset to the network which contained the 00001 contact. By subsequent multiple retrace commands, these locations are found again in reverse order. If the nesting level exceeds 20, the first stored location is deleted, and all other ones are shifted by 1. The shorthand key is <**Alt B**>.

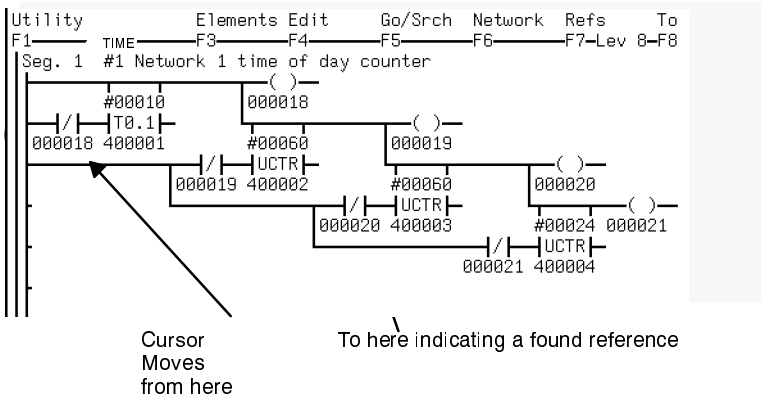


Figure 216 Trace Example

6.5.1.5 Show Trace/Search

When executing this function, a window is opened on the screen showing the actual Trace and Search list (as previously illustrated). This list contains the following information:

- Reference Number from which Trace or Search was called
- Segment Number of Location Trace or Search was called
- Step # of Location Trace or Search was called
- Network Number of Location Trace or Search was called.

- Trace History data base can have 20 entries.

After pressing any key, this window is removed from the screen.

6.5.1.6 Delete Trace/Search

When starting a new trace or Search, the contents of the trace list can be deleted by the function Delete Trace/Search. When used, all locations previously stored are deleted.

6.5.1.7 On-Line Search

If you are Online, the search is controlled by the PLC and not the panel software and searching for a reference in a function block will fail. Modsoft prevents this type of online search and posts a message if the search is attempted. To search for references that include function block definitions, use Modsoft.

6.6 Network

6.6.1 Insert Before Network

This command inserts a new network before the current one. The current network and all that follow are shifted to the end, renumbered, and the inserted empty network is shown on the screen. For the inserted network, an empty title and comment area are inserted. The shorthand key is
<**Alt I**>.

6.6.1.1 Insert After Network

This command inserts an empty network after the current one. This new network is displayed on the screen after execution of this command. Identification management is similar to the Insert function above. The shorthand key is <**Alt A**>.

6.6.1.2 Copy Network

You can select this function to make a copy of a network complete with associated comments. After selecting copy, use other commands such as PgUp and PgDn to select the destination location then use the paste function to complete the copy.

6.6.1.3 Delete Network

This command deletes a complete network, including the network comment. The command places the deleted network in a buffer, which can be restored by using the paste network selection. This is the same paste buffer used in the Network List Editor. This Paste feature is useful if you accidentally select the wrong network to delete.

6.6.1.4 Paste Network

Use this function to restore a deleted network or to duplicate a network. First delete or copy the network of interest then position the cursor at the row that the network should appear in and press the < ↵ >.

6.6.1.5 Equation Network

An Equation Network is a specific functional Ladder Network that you design and implement as a programming element using this selection and a dedicated Editor. Equation development is presented in detail in Chapter 7.

6.7 Refs

The Reference Pulldown is slightly different between Offline and Online programming modes where:

- **Offline**
  Screen Size [Alt W]
  Reference Zoom [*]
  DX Zoom[Alt Z]

- **Online/Combined**
  Screen Size [Alt W]
  Reference Zoom [*]
  DX Zoom[Alt Z]
  State Flow
  Used References
  To/From Level 0

6.7.1 Screen Size

Screen Size is a toggle for display expansion as defined in the Reference Editor Chapter.

6.7.2 Reference Zoom

Specific logic nodes can be “zoomed” to by placing the screen cursor on a node and pressing the shorthand ***** key. A window displaying the data for that node is displayed. If the node reference is a relay or coil, the disabled/enabled state is displayed. The value field shows the current value of that reference. The I/O Map reference shows whether the reference is configured in the program’s I/O Map, where Yes is configured and No is not configured.

The Symbol area can be edited with the “S” key. The Descriptor area can be edited with the “D” key.

The disabled/enabled state of a contact can also be edited by pressing the <**E**> key. Likewise, press the <**V**> key to change the value field.

These illustrations provide an example of reference zoom for the first element in the ladder diagram and the field structure of the data base.

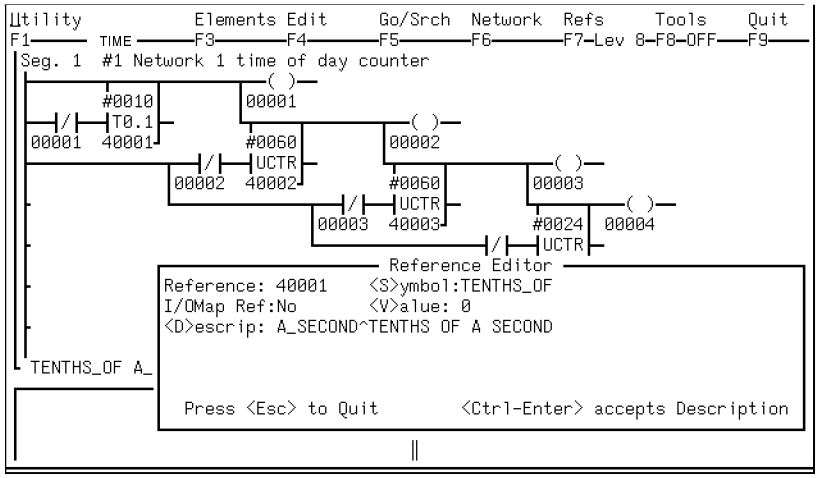


Figure 217 Example of Reference Zoom

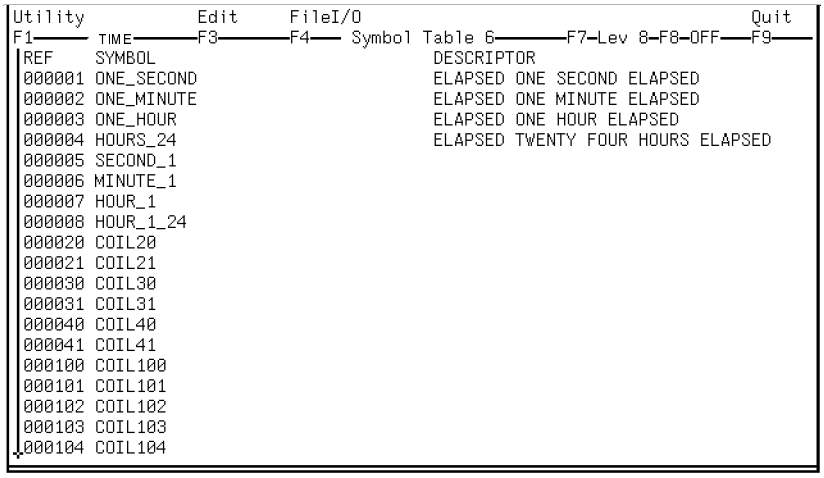


Figure 218 Reference/SymbolFields

6.7.3 Dx Zoom

Dx functions that are included in the Zoom template allow the programmer to review the data relative to the function. The display of a PID2 for example, illustrates the data presentation and the ability to change format between Hexadecimal, Decimal and Binary.

In Dx Zoom, registers are displayed in a predefined multi screen format which allows users to

edit the contents and display current values within the Dx Block. Comments are provided to assist in determining what each register represents.

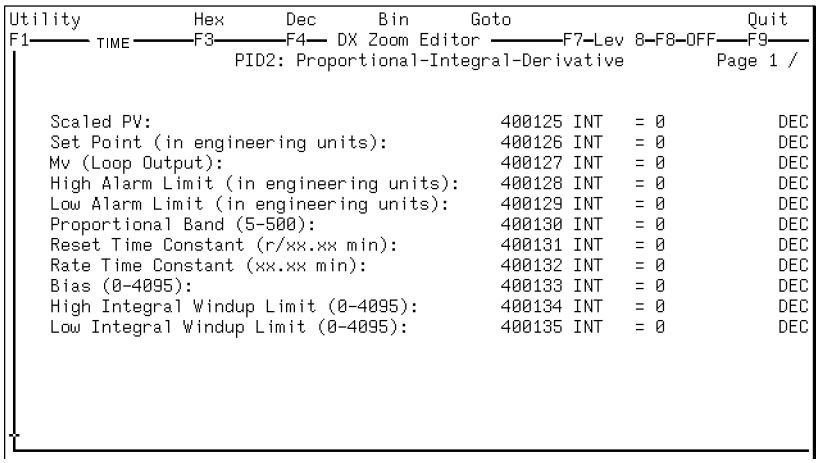


Figure 219 PID2 DX Zoom example

- DX Zoom Supported Functions

CALL	MAP3	STAT	DMTH
MATH	WRIT	DRUM	MRTM
XMR	EARS	MSTR	XMWT
EMTH	MBUS	DIOH	EUCA
PCFL	HLTH	PID2	HSBY
PEER	ICMP	READ	IMIO
ITMR	CHS	IMOD	

6.7.4 State Flow (Online / Combined Only)

State Flow enhances the traditional power flow i.e., current path created by contact closure by testing the logical status of all normally open or closed contacts in a network to indicate whether the design would provide a power flow path or not.

To illustrate this concept, a simple diagram with a single instance of state flow is presented. This diagram is illustrated below and its initial online status shows “Power Flow” at normally closed contact 00130. Both Power flow and State flow are indicated by a higher intensity display at the element.

By setting input 10001 to an On condition, there is no change in Power Flow but if you select the State flow display, you will see the initial power flow AND State flow indicated at the point 10001.

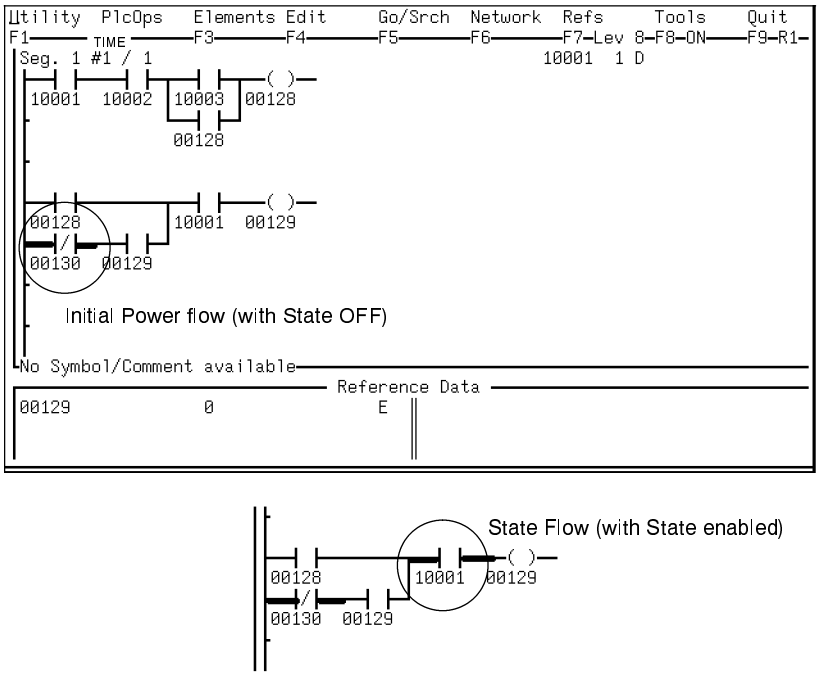


Figure 220 Network 1 Power and State Flow

The statement made by the above example is; if coil 00129 is activated the logical state of input 10001 will be ON.

6.7.5 Used References (Online and Combined)

The last item in the Direct to PLC "`Refs`" pulldown is an Used Reference selection that allow you to map already used references while you are developing/debugging a program.

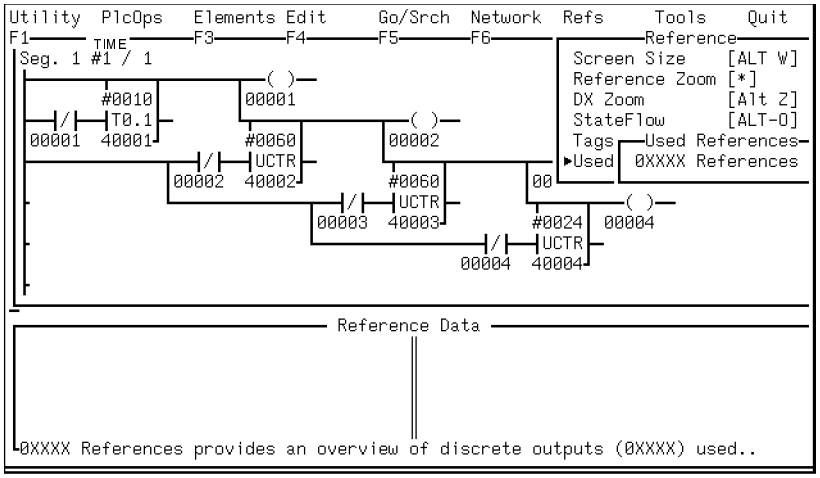


Figure 221 Online Used Reference Selection

The following is an example of the 0X map using the “time” example for data. In Online mode, only the 0X reference is mapped. The Combined Mode offers 0, 1, 3 and 4X used reference maps. If you have elected to use the /u switch used references are not mapped and the program conversion time is faster.

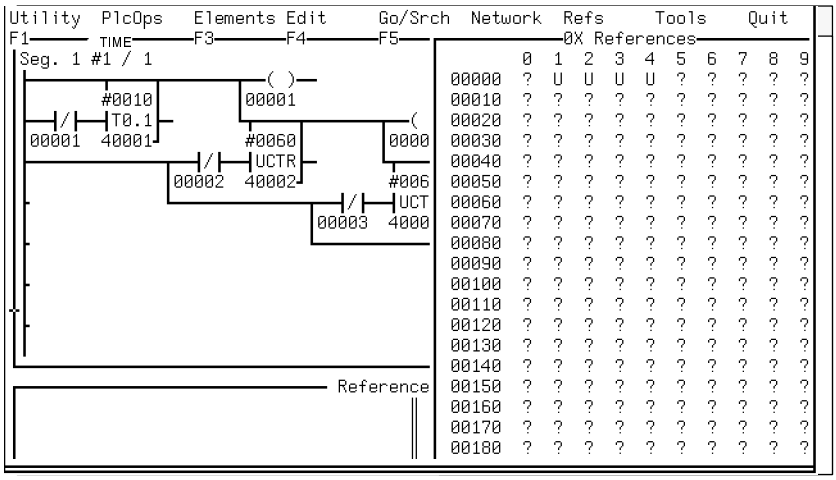


Figure 222 Used Reference menu selected

Use of a reference is displayed according to the following legend.

Highlighted dash **-** I/O Mapped

U	Used Coils
L	Accessed in Ladder
?	Not used but may be referenced
-	Not used in Ladder
Highlighted **U** or **L**	I/O Mapped and used in Ladder.
Highlighted **?**	I/O Mapped but may be referenced in Online program logic

6.7.6 To/From Level 0

This switch is available in Online or Combined modes and allows you to toggle between the user level set as your default and User Level 0. If Level 0 is the default, the toggle is not selectable.

While in the Level 0 mode,you can edit ladder or other selected editor data but when you exit the ladder editor the Original User Level protection setting is reverted to. Each subsequent use of the ladder editor will be entered with the last online level you set.

6.8 Tools

Most of the functions available in the Tools pulldown, are described and located in other Menu functions but are gathered here for the programmers' convenience. Availability between programming modes is:

- **Offline**
  Network Comment [Alt C]
  Subroutine Segment
  File Check
  I/O Map
  Configuration [Ctrl F8]
  ASCII
  Save Changes [Ctrl F4]

- **Online/Combined**
  Network Comment [Alt C]
  Subroutine Segment
  I/O Map
  Configuration [Ctrl F8]
  ASCII
  Save Changes [Ctrl F4] (Combo only not Online)

6.8.1 Network Comment

This editor is a special text editor that lets you edit the title and comment of a network. When a network is inserted in the program, an empty title and comment are automatically assigned. The shorthand key is <**Alt C**>. Refer to Chapter 4 for details.

☞ **Note** When editing the body of the comment, if you use "`delete`" to remove a line, the line becomes blank. To close up the file, use `delete` again to cut the remaining lines and paste them at the deleted blank line.

6.8.2 Subroutine Segment

Pressing this select moves the user to the last configured segment. With initial defaults, the Micro user is set to to segment 2. The remaining PLC's are set to 32. The subroutine segment is where you use the DX subroutine handlers RET and LAB. This segment is not solved until the scan finds a JSR (Jump to Subroutine) and executes it. The JSR then calls the appropri-

ate LAB, executes the logic until the RET (Return) is reached. The RET resumes the solve just after the JSR block and continues logic solve from that point.

6.8.3 File Check (Offline Only)

This function allows you to:

- Compare the Checksum of program logic and the first 128 words of configuration to determine if it matches the configuration it is based on.

6.8.4 I/O Map

The status of I/O modules can be displayed from both the Online,Combined and Offline modes when the menu item "I/O Map" has been selected. If the controller is stopped, the same window is displayed, with the last column (description) used differently. The information is taken out of the controller's configuration table. The right of the screen shows the modules physically put into their corresponding slots, so you can compare the configured modules with those actually available. If the controller is running, a reduced display only shows whether the I/O module health is OK, or not OK as indicated by an * beside the Module ID (the full I/O module status is accessible only if the controller is stopped). The values in the I/O can be changed as in the Offline Mode.

Caution On a Compact 984 adding modules that require a loadable driver, the controller will not start because the loadable cannot be added Online. You must go offline to add the new loadable driver to the controller configuration.

6.8.5 Configuration

When selected, controller configuration is shown on the screen. The configuration menu in Online mode only presents the "I/O Map", "Ports", "Loadable", "Configuration Extension" and "Quit". All changes within the configuration have to be made with the controller in Stop mode.

6.8.6 ASCII

Selecting ASCII puts you in the ASCII editor display. See Chapter 9 for details on ASCII editing.

6.8.7 Save Changes (Offline or Combo Only)

This entry provides the same function as offered and described under the Offline Menu Save Changes function Defined in Chapter 3.

6.9 PLC Ops (Online and Combo Only)

The Ladder Editor menu is only available if you are in the Online and Combined mode.

6.9.1 Start/Stop

From these selections you can start or stop the PLC logic execution.

6.9.2 Single Sweep

Allows you to do one or more single sweeps. The controller is started before running the first scan; when you select single sweep a display request is provided for you to enter a Time-out range (010-200) where the default is 10 milliseconds. The PLC Status screen will now indicate the operating mode as “Single Sweep”.

As you monitor an active Network in Online Mode, the Segment Status display (bottom Line) prompts you to Press <**Ctrl F9**> for a next sweep or <**Ctrl F10**> to end sweeps. When the choice is made to end a sweep, the network returns to continuous scan functionality.

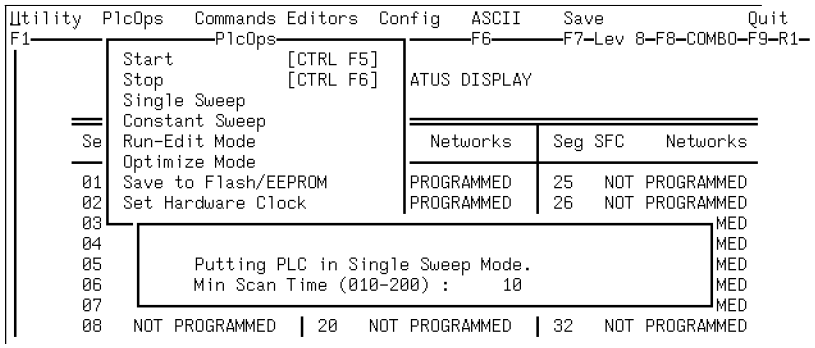


Figure 223 Single Sweep Selected

6.9.3 Constant Sweep

The constant sweep mode can be called if the controller is started. When selected a prompt is displayed for you to enter the proper values. The parameters remain until constant sweep is turned off.

After entering the parameters the display notifies you that Constant Sweep is on. If you re-en-

ter constant sweep from the Comm menu, Modsoft displays the prompt:

Finish Constant Sweep? (Y/N) _

If you choose Y the Prompt **Constant Sweep : OFF i**s displayed.

The Single sweep scan message is displayed on the bottom of the Segment Status display.

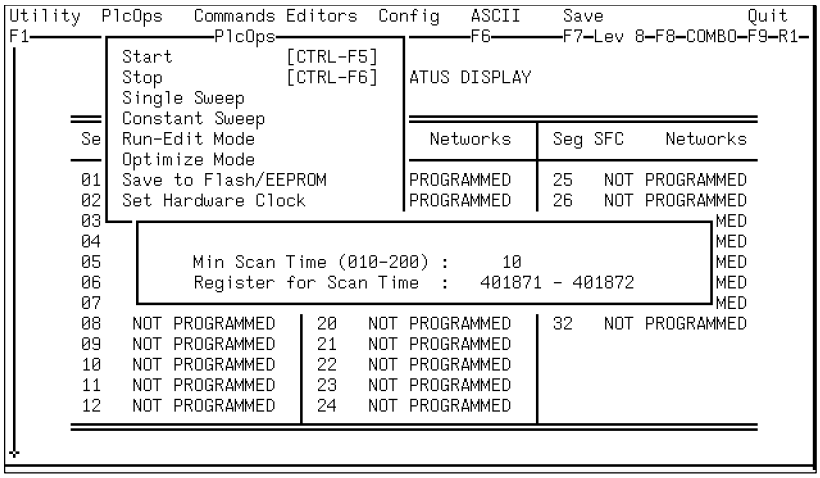


Figure 224 Constant Sweep Parameter Entry Screen

6.9.4 Run- Edit (PLC Stopped)

This command conditions the way the PLC will run the next time it is started. To execute the command you must be Online and the Controller must be "stopped". Run-Edit is the normal operating condition for the controller. This mode searches for the disabled states of coils during each scan (Optimize does not).

6.9.5 Optimize (PLC Stopped)

Optimize mode is the fastest controller operating mode. The function is intended to save logic solve time by not searching for disabled states of coils. To start a controller in this mode, discrete points are not allowed to be disabled and power flow is not displayed on the programming panel. Changes to the program are not allowed while running.

6.9.6 Clear Logic (PLC Stopped)

Clears the PLC program and the State RAM database. The application automatically returns you to the Offline Main menu.

6.9.7 Save To Flash/EEPROM (Controller Stopped)

This function is active when online to transfer a program to ***those controllers in the Schneider Automation, Inc. Product Line, that support either the EEPROM or Flash memory technology as is the case on the Compact, M1, Micro and 434/534 Quantum-controllers***.

> **Note** The program to be downloaded may not include any USL (user) loadables. (.exe)

When you select this function from the `PlcOps` pulldown a submenu is displayed to prompt you for Enhanced EEPROM Loading Options.

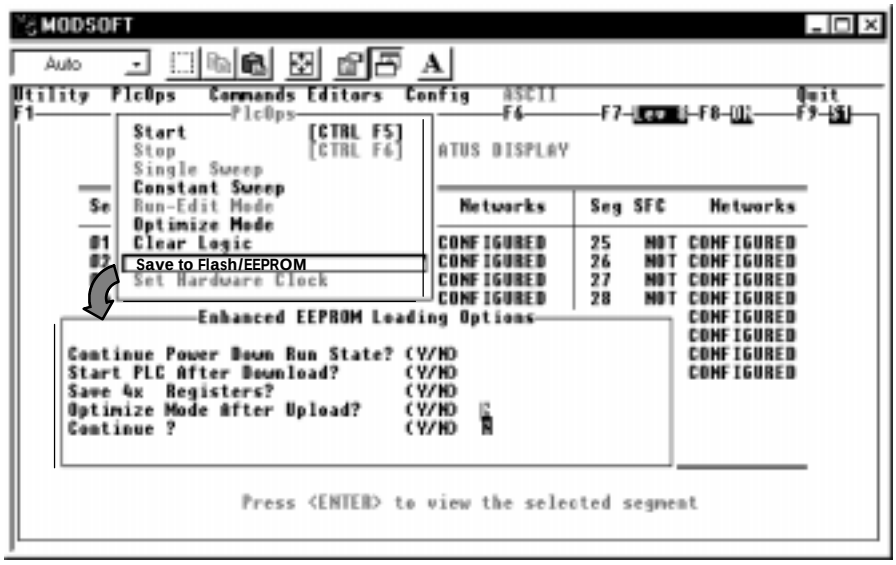


Figure 225 Store to Prom/Flash Selected

These are the selection Options for Compact and Magnum:

- Continue Power Down Run State? Y/N
  Causes the controller to run/not run based on the run state at the time the controller was powered down. This requires battery RAM retention prior to powering down. If the battery is missing or discharged, the controller will not auto start on power up, regardless of the power-down state.

- Start PLC After Download ? Y/N
  Causes the controller to go into run mode after reloading from the EEPROM/Flash. This is over-ridden if the user selects YES for the Continue Power Down Run State.

- Save 4X Registers ? Y/N
  This selection causes the PLC to save the current 4x registers contents in EEPROM/Flash. This will *not* maintain the *disable* states of contacts and coils.

- Optimize Mode After Upload ? Y/N
  Allows you to select either optimized or non-optimized mode on startup from MEEP. This selection allows user logic editing after download, *if optimized mode is not selected*

- Continue? Y/N
  Enter “Y”to start the save to Flash operation. None of the above answers can be blank. All must have either the Y or N.

Of the above options, the “Save 4x registers” and “Optimize” choices are not applicable to the Momentum M1 controllers and are not available.

The System Message will indicate a successful transfer. In the event an error occurs an error message is displayed.

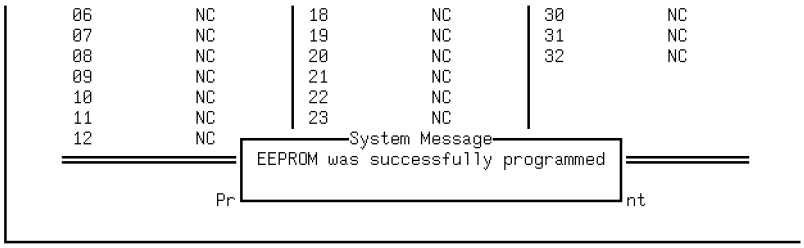


Figure 226 Prom/Flash Transfer Successful Message

If you have a Micro model configured you will see the following

Flash RAM Loading Options
Automatically START PLC if the program
is loaded from Flash RAM Memory ? y/n
Save PLC contents in Flash Ram Memory ? y/n

☞ **Note** All questions must be answered Y or N for transfer to take place.

6.9.8 Set Hardware Clock

The Online Ladder Diagram `PlcOps` menu has a pulldown item called `Set HardwareClock`, which lets you set the hardware clock on the connected controller. Using this command, you are asked for date and time input in USA convention as:

Date: mm/dd/yy Time: hh:mm:ss

The actual controller date and time information - the contents of the Time/Date register - is offered as a default value. You can modify both of the default values. The modified values are checked to see whether they represent a valid time and date. If not, an error message is displayed. Pressing <**Enter**> writes the change to the controller. The Time of Day clock can only be set if you have configured it in Specials and transferred the configuration file to the PLC.

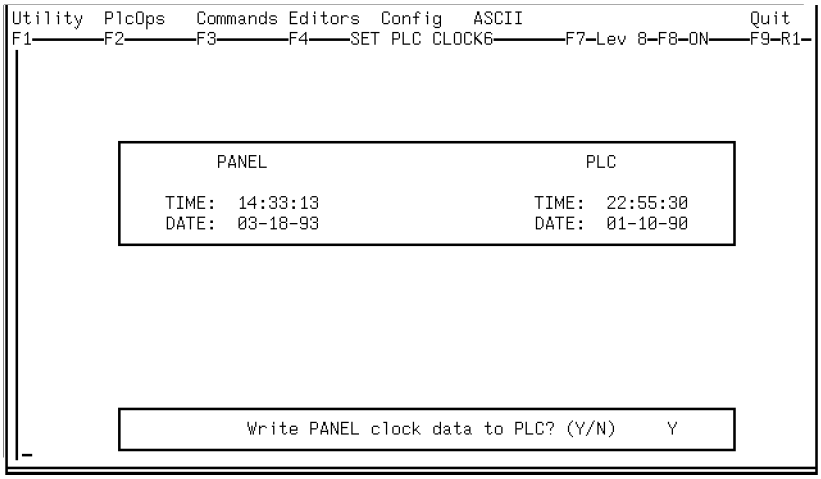


Figure 227 Hardware Clock Interface Display

6.10 Online/Combo Protected Edits

6.10.1 Ladder “Refs” Menu

When in Online or Combo mode, you have additional entries in the `Refs` pulldown:

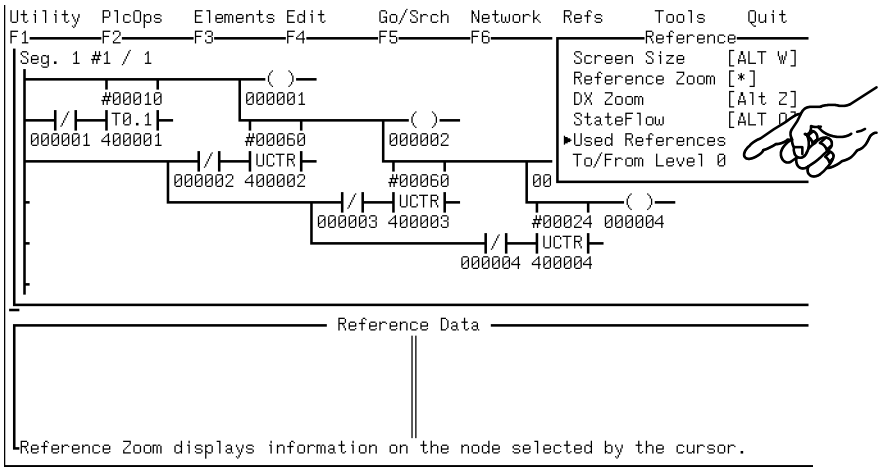


Figure 228 Protected Edit Menu Item

The `To/From Level 0` menu item is disabled (grayed out) if your start-up setting is already at 0. Additionally in offline the item is not available for selection.

When selected the user level toggles between level 0 and the level you have previously set and is indicated between the F7 and F8 on the Modsoft function key line. The level stays at 0 for other editors that are presented within the ladder editor; however when you exit the ladder editor the initial setting is returned.

Chapter7
Equation Network Programming

□ This chapter instructs you on how to implement an Equation Network in programs intended to be downloaded to Controllers that support them.

7.1 Equation Network Overview

This Chapter covers the Equation network function block that is part of the Modsoft version upgrade 2.3.

The Equation function highlights are:

- A special Equation type network in the ladder logic with the ability to specify the value of an output register in algebraic notation.
- The ability to enter equations as combinations of the standard math operators (+ - * / **), all of the math functions allowed in the EQN block within PCFL plus conditional and logical expressions, with the further ability to specify variables and constants as necessary to the calculation and to group the expressions with parentheses.
- The ability to use reference variables and constants of any type in these equations by specifying the type.
- The ability to save and manipulate these new networks similarly to ladder logic networks.
- The ability to print these networks along with the other networks.
- Used Refs for the coils and registers in this new network type will operate as they apply to coils and registers in usual ladder logic networks.
- Limited ability to do Search and Trace. Search and Trace operations begun in regular ladder logic networks will be able to find references in Equation Networks. Similarly, elements within the Equation Box are not be able to be selected for Trace references that occur within the Equation Box. Searching online/combo can find all horizontal shorts except for those in the Equation editor.

Note You can use an Equation in a Macro but Macro commands are not allowed within the Equation

7.2 Support Requirements

Equation Networks are currently supported only by Quantum PLC's with executive ID of 871 or 882.

Program files containing Equation Networks are supported by Modsoft versions 2.3 and above.

If a program containing Equation Networks is downloaded to a PLC that doesn't support Equation Networks, Modsoft will display an "illegal node" error message.

Floating Point hardware is not required. However it improves the scan rate for equation networks with many floating point calculations.

7.2.1 Product Features

All the features of Equation Networks depend on the ladder logic implementation of this new network type. The areas of operation that are affected are:

- Network Editing

  Reference and Symbol entry
  Search and Trace
  Network Printing
  Network transfer to the PLC of a program file

  Network Editing further divides into:
  the Menu bar,
  the Equation Network display,
  the Equation editing window in regular and wide mode
  (wide mode not initially supported)

The Equation Network is another level of ladder logic. Equation Networks are entered and edited from the ladder logic portion of Modsoft.

7.2.2 Equation Network Editing

The Equation Network is the means of providing you with the Equation capability in a ladder context. The ability to write an equation, that has similar flexibility to that of a high level programming language such as Pascal or C, and integrate this ability into the rest of ladder logic, requires a ladder logic interface to the Equation language. This new network, called an Equation Network, has a limited set of ladder logic elements and an Equation Element. The Equa-

tion Element is a statement in Equation. How this new network is created and how it is treated by Modsoft at the highest level is menu driven just as the overall Modsoft interface is.

The most important change is to the Network submenu. There is one new choice so that, if the PLC is a Quantum, it will appear but if the PLC is not a Quantum, it will not appear.

Insert Before Network

Insert After Network (append)

Copy Network

Delete Network

Paste Network

Equation Network

`Equation Network` must be chosen *in the ladder logic editor* for an **empty network**, then the empty network is transformed into an Equation network. If `Equation Network` is chosen for a network with any contents, that is, the network displayed has ladder logic elements, an error dialog box appears and no equation network is created.

In an Equation network, insert/append network will create another Equation network.

7.2.3 Menu Bar

The **UTILITY** Submenu

The Utility submenu functions no differently for Equation Networks than for other ladder logic networks. If processing is in an equation network, key help is appropriate.

The **PLC Ops** Submenu

The **PLC Ops** submenu, which appears in Online or Combined mode, functions no differently for Equation Networks than for other ladder logic Networks.

An Equation network can be defined Offline, or a program that contains an equation network can be selected to be “read in” for editing as long as the destination PLC is defined to be a Quantum.

The **Elements** Submenu

The **Elements** submenu has all but its Relays selection grayed and un-selectable.

Relays only functions for the specific reserved locations of the Enable contact and the output coils where the contacts are limited to normally open and normally closed (up transition and down transition are not allowed) and horizontal short/horizontal open. The coils are limited to a simple coil (no latched (memory) coils are allowed). The shorthand keys for the allowed elements are allowed. The shorthand keys for non-allowed elements is not allowed (i.e. will be ignored). The Enable contact can be any 0X or 1X. The output coils may be any 0Xs.

The **Edit** Submenu

An equation can be cut, deleted, copied, or pasted only as a whole. It can be pasted into another Equation Network, then edited using offset or element by element. The Equation Box cannot be cut, deleted, copied or pasted piecemeal. Functions that can be selected from the Network submenu but are not allowed result in the display of an appropriate error message.

Offset works similarly in equation and ladder logic networks, except that in equation networks the range of offset is always the entire equation.

The **Go/Srch** Submenu

In the **Go/Srch** menu, all of the functions should operate on equation networks just as they do in ladder logic networks with the limitation that the specific reference found inside an Equation Box is not highlighted or its multiple occurrences noted. Similarly a Trace or Retrace operation called outside an Equation Network displays the appropriate Equation Network but not the reference within an Equation Box and a Trace operation cannot be called for a reference inside the Equation Box.

The **Network** Submenu

The **Network** menu works the same as the regular ladder logic networks. It manipulates whole networks. Only when a paste of an equation network is desired, do you have to do anything special. Specifically, to paste an Equation network, a new, empty network must be inserted and declared to be an Equation Network. Then, inside the Equation Editor, you may paste the current cut or copied Equation Network. Otherwise, it does not appear that anything special is happening. The added function to define an empty network as an equation network is the only indication of difference.

The additional selection, Equation Network, does not appear (or will always be greyed) if the PLC is not a Quantum. When you are in the ladder logic editor, this menu choice defines an empty equation network and goes into the equation network editor. If you are not in the ladder editor this menu item is not selectable.

The **Refs** Submenu

The **Refs** menu which differs between `Offline`, `Online`, or `Combined`, also differs depending on whether the current network is an equation or ladder logic network. **DX Zoom** is un-selectable if the current network is an Equation Network.

Used References - this function works as it does for regular ladder logic.

The **Tools** Submenu

The **Tools** menu which differs depending on whether we are offline, online, or combined, will not differ depending on whether the current network is an equation or ladder logic network. The only selection which will require functionality that will differ for Equation Networks is **Save Changes** A program file language necessary to represent equation networks is developed and the translation routines are be enhanced to handle it.

7.2.4 Equation Network Display

The behavior of the ladder logic editor when the current network is an Equation Network differs in these ways:

- the network template has a different appearance,
- there are limitations on the 6 ladder logic nodes that are allowed in an equation network
- the Equation element is treated as a separate editing window

Except for the Help function, once the Equation element, (also called the Equation Box), is opened for editing, no other menu selection or Modsoft operation can occur until the Equation Box is closed.

The initial screen appearance for an Equation Network is:

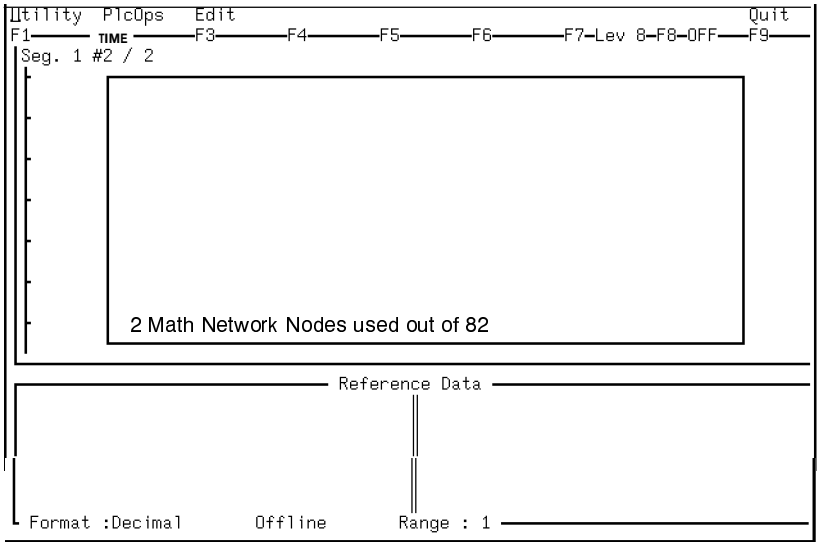


Figure 229 Initial Display for Equation Network

For this element and for the other elements used in the Equation Network, the rules for specifying their input, output or register references are the same as for entering or editing ladder logic elements. The arrow keys move about the equation network except when there are no elements in the indicated direction to move to, in which case they wrap or move into the Reference Data editor, again similar to regular ladder logic editing.

You move out of the equation area by pressing the <**Enter**> Key then cursor to each output element by pressing the proper arrow key. To move to the Input enable element from the output area press the <**Right**> Arrow key (this is essentially a wrap operation).

The five optional output coils, in order, top to bottom are:

Done
Lesser
Zero
Greater
Error

The ladder logic elements become highlighted when selected and allow entry of their elements and references just as in normal ladder logic (with the restrictions on which elements can be specified). If any key but an arrow key is entered when the equation element is se-

lected, the equation window is opened for editing. Thus the seven elements of an Equation Network are: Enable, Equation, Done, Lesser, Zero, Greater, and Error.

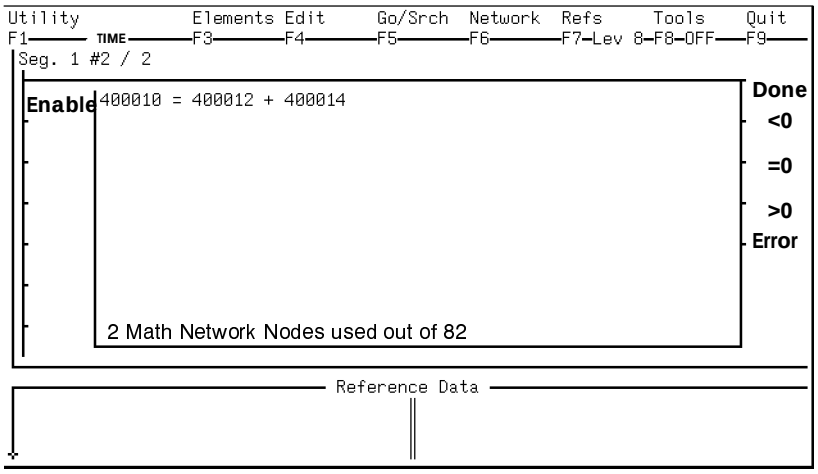


Figure 230 Menu and Display after Equation Entry

If an Equation Network already has contents they are displayed in the equation box (and the equation element is not in edit mode). In online or combined mode, if the PLC is running, the Equation Network's enable element and output coil elements are highlighted according to the power flow reports. If there is not yet a legal Equation Element, the equation is not present in the PLC and is not being calculated and thus none of the output coils will receive power.

If you want to see an example of a simple, but complete equation network you can load a program from c:\modsoft\programs\TIME_EQN that appears like:

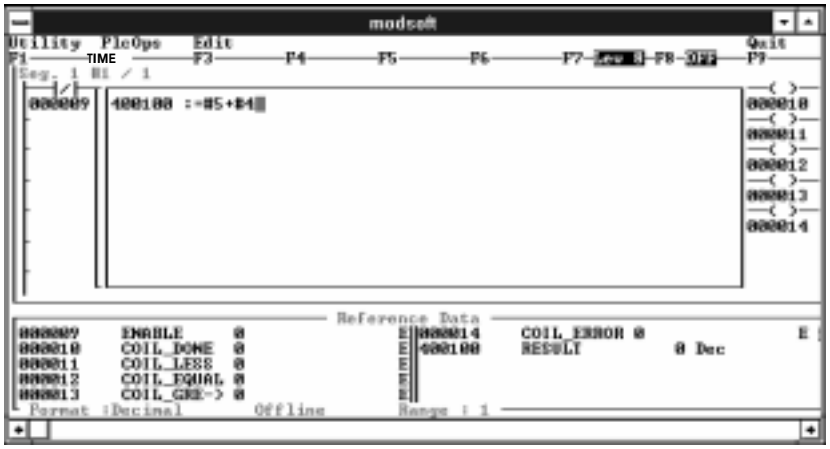


Figure 231 TIME_EQN Equation Example

7.2.5 Equation Editing

See: Equation Networks in 840-USE-101-00 The Modicon Ladder Logic Block Library.

At the Network level, the equation box is a single element. When the Equation Box is activated for entry/edit it has its own collection of elements. Once activated, the user can move around in the equation box, insert elements and delete elements by simple ASCII character editing. This existing edit mode has both insert and over strike modes. The keys have the following functions:

Key	Operation
bkspc	delete previous character
del	delete the current (highlighted) character, moves succeeding characters left so that the next character is moved into the cursor position
←	move the cursor back over a character
→	move cursor forward over a character
<char>	in insert mode, insert the typed character before the current character and move the cursor forward (so it remains on the same character); in overstrike mode, replace the current character with the typed character and move the cursor forward to the next character
<enter>	parse equation and close Equation Box if equation legal.
home	move cursor back to the first character
end	move cursor forward past the last character
<esc>	restore previous contents of equation box (if any) and close it
ins	in insert mode, toggle to overstrike mode; in overstrike mode, toggle to insert mode

Since there is overstrike mode in equation editing, characters may be deleted by explicitly deleting them or by overwriting them.

The equation is treated as a string of text. It automatically wraps lines but individual elements of the text are only recognized after you have typed the <enter> key ↵. At this point the string is translated into elements (tokens), that are validated individually, and the string is put in canonical form. The canonical form is the same form that would be read in from a program file or uploaded from the PLC. Tokens are not split between lines, and are separated from the others by spaces, implied types will appear explicitly. This form indicates to you how the equation has been interpreted.

If an error is detected when the equation is parsed, the cursor is placed at the start of the token that is a problem, an error dialog box appears, and the equation remains open for editing. If none of the tokens cause a problem, the string is parsed as a whole and if there are any problems, the cursor is placed at the point at which a difficulty was found and the same events occur as for a token error. If there is no problem with the equation, the equation element is closed and saved to the current program file, downloaded to the PLC or both depending on the mode of operation current: Offline, Online or Combined.

7.2.6 Equation Elements

When the equation is opened for editing, the cursor is placed on the first character in the equation and it is highlighted if the equation has any contents.

The first element in the Equation is its Output, which must be a 0x, 4x reference and must be 'typed' or is assumed to be 16 bit signed integer.

After the output element is the equation itself.

Elements in the Equation are:

constants, references, operators, functions, and parentheses.

These elements can be combined to form expressions which then act as elements in larger expressions. That is, the expression:

#2.0 * (300010F + 300014F)

has the expression 300010F + 300014F acting as an element in a larger expression. Parentheses are not always necessary when combining expressions unless order of evaluation needs to be specified. In the example above, 300010F + 300014F is calculated before the

multiplication. In the example below the order is unimportant so no parentheses are used, yet an expression is combined with a binary operator and another expression to form a larger expression. (In cases where 400014 and 300010 are both very large, division should be done first, and should be forced by parentheses.)

```
400002L * 400014L / 300010F
```

7.2.7 Lexical definition

An expression can be something as simple as just a register or a constant or it can be quite complex. For example, 300010 (a register), #23U (a constant), (300010 * #23U) and (400021U & #27U) are all expressions.

The first two are examples of the simple expression, the last two are of binary operator expressions between two simple expressions.

A third type of expression is a conditional expression. Conditional expressions are complex expressions. They contain a relational expression followed by '?' followed by two choices associated with TRUE and FALSE results of the relational expression. For example:

```
400001 = 400021U & #27U ? 300045 : 300055
```

It is possible to have embedded relational expressions in expressions.

Thus: (400021U & #27U) is evaluated to TRUE or FALSE, as it combines the contents of register 400021 and (in binary format), 0000000000011011, yields a value which is the bitwise "and"ing of the bits in the register with the bits in the constant. The expression is evaluated as TRUE if any bits are "ones" in the result and false if all bits are "zeros". In this example, if the result is TRUE, 400001 is set equal to the contents of the input register 300045, and if FALSE, to the input 300055.

7.2.7.1 Operators and Functions

binop : Arithmetic operator
| Relational operator
| Logical operator ;

Arithmetic operators:
+ (plus)
- (minus)
* (multiply)
/ (divide)
** (raise to power)

Relational operators :
< (less than)
<= (less or equal)

Logical operators :
& (and)
| (or)

= (equal)	~ (1's Complement)
<> (not equal)	^ (exclusive or)
>= (equal or greater)	>> (shift right)
> (greater than)	<< (shift left)

Functions :

ABS	(absolute value)	SQRT	(square root)
EXP	(power of 'e')	SIN	(sine(radian))
LN	(natural log)	COS	(cosine(radian))
LOG	(log base ten)	TAN	(tangent(radian))
SIND	(sine(degree))	COSD	(cosine(degree))
TAND	(tangent(degree))	ARCSIN	(arc-sine(radian))
ARCCOS	(arc-cosine(radian))	ARCTAN	(arc-tangent(radian))
FLOAT	(change fixed to float)	FIX	(change float to fixed)

☞ **Note:** In modsoft, the functions require to be entered in all capital letters as shown above. Lower case will not work.

Constants: All constants are in decimal, (Hexadecimal is not allowed).

Math format types: References and constants must have types. The types are as follows:

U Unsigned short (16 bits)

S Signed Short (default type)

L Signed Long integer (32 bits)

UL Unsigned Long integer (32 bits)

F Floating point number (32 bits)

B binary (1 bit) (default for discrete references)

☞ **Note:** Binary Type is not allowed on 3x or 4x registers

Note: In modsoft, these Type characters must be in Upper case.

Examples of Expressions:

300012 > 400041

The value is 1 if 300012 is larger than 400041, otherwise 0

400031 & 400027

The value is the bitwise AND of the contents of 400031 with 400027

(100022 ^ 000022) & 100023

The value is a bit since all the operands are bits, if 100023 is 0 the value is 0, otherwise the value is 1 if 100022 and 000022 are different, 0 if they are the same.

400041 | (100022 & 100023)

the value is the result of OR'ing two expressions whose values are bits, and is 1 if either expression is 1, otherwise 0.

400035F > 400033F ? (300033F / 400035F) :

(400033F> #0.0F ? (300033F / 400033F) : 300033F)

The value is a floating point number, the result of an if that contains an if. If the floating point number in 400035,400036 is larger than that in 400033,400034, 300033,300034 is divided by 400035,400036 and this forms the result. However, if 400033,400034 is equal to or greater than 400035,400036, the alternate expression is computed to obtain the result. In the alternate, if the floating point number in 400033,400034 is greater than 0., then 300033,300034 is divided by it, but if 400033,400034 is equal to or less than 0., the result is simply 300033,300034. That is, there are three possible results depending on the values of the floating point numbers starting in 400035 and 400033, they are; 300033/400035, 300033/400033, or 300033.

7.2.7.2 Operator Precedence

The operations which are performed before other operations in a long string of variables and operators, are determined by operator precedence. The precedence used by the PLC for equations is:

Operation Group	Operations
Unary	- ~
Exponentiation	**

Multiplication	*
	/
Addition	+
	-
Logical	&
	|
	<<
	>>
	^
Comparison	<
	<=
	=
	>=
	<>
	>

This is best illustrated by the expression:

300001 ** 300002 * 300003 + 300004 & 300005 > 300006

which is evaluated by just doing the operations as they occur, first 300001 ** 300002 then the result of that * 300003 then the result of that + 300004 then ANDing the result of that with 300005, then comparing the result of that with 300006, which would turn the value into a single bit TRUE or FALSE. That is, it would be what could be indicated by parentheses as:

((((300001 ** 300002) * 300003) + 300004) & 300005) > 300006.

Operator precedence would have the opposite effect on the expression

300002 > 300003 & 300004 + 300005 * 300006 ** 300007

which would be evaluated in the opposite order to that in which it is written, starting at the right with 300006 ** 300007, as we would indicate with parentheses as:

300002 > (300003 & (300004 + (300005 * (300006 ** 300007))))

One more example of operator precedence:

300001 < (300002 | 300002 & 300001 + 300003)

On precedence alone it would be evaluated as if it were parenthesized:

300001 < (300002 | 300003 & **(**300001 + 300003 **)**)

which may not be the user's intention which perhaps should have been parenthesized:

300001 < (300002 | 300003 & 300001) + 300003

Note: The results of these two statements are not equivalent, so you must be careful to specify the exact meaning intended.

7.3 Reference Entry

Output 4x references are legal for output.

Input 1x or 3x references are input only, and may not be on the left side of an equation.

Discretes 0x and 1x references are called discrete points. They carry single bit binary values. If a discrete is an object of an expression, the result must be a single bit (0,1).

Reference Lengtth All references should be entered in either the 5 char or 6 char length modes, i.e. 40001 or 400001. Both are legal in Equation parsing. Of course, entering a reference above that which is configured for the controller will result in an error, as ever.

Constants Numbers beginning with '#'. All constants are in decimal,

Math Format Types References and constants must have types. The types are as follows:

U Unsigned short (16 bits)
S Signed Short (default type)
L Signed Long integer (32 bits)
UL Unsigned Long integer (32 bits)
F Floating point number (32 bits)
B Binary (1 bit) (default for discrete references)

Default types: any register or constant in an equation used without a Type indicator is assumed to be a signed short. Any discrete reference is assumed to be a Binary .A binary constant must have the 'B' type suffix; only "0B" and "1B" are legal.

7.3.1 Reference Examples:

300005 is type S, the default for unspecified registers 3x, 4x, and.

000003 is type B, the default for unspecified registers 0x and 1x.

400012F is type F and as a 32-bit floating type, includes 400013.

300021L is type L and as a 32-bit integer type includes 300022.

400010B is not legal since a 4x register is 16 bits.

000022U and 100003F are also not legal.

7.3.2 Constant Examples:

#65535U The largest unsigned short allowed.

#123456U Illegal, too large for the type declared.

#123456L and #-9876543L Signed longs.

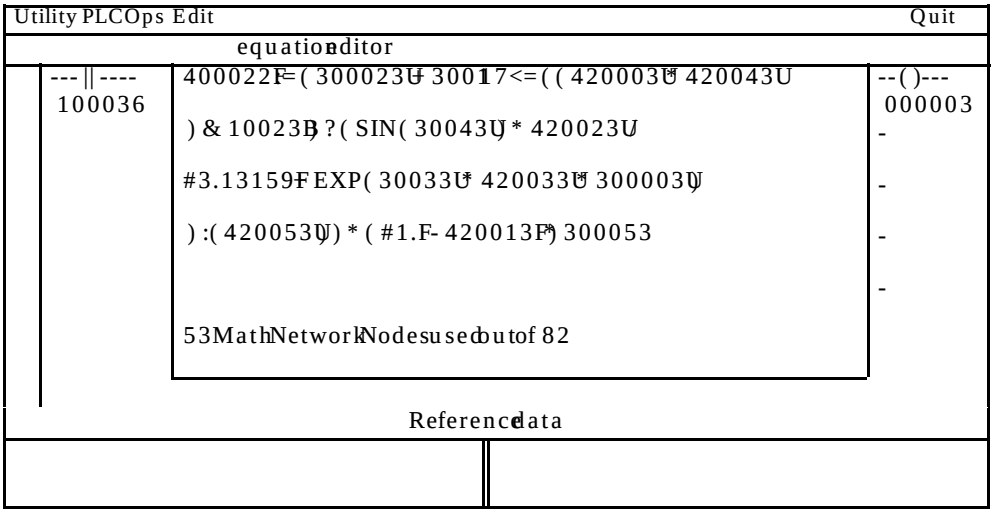
#1.175494 E-38F and #3.402823 E+38F Limiting values of floating point constants.

#3.14159F Floating value. (Pi)

Note: The reference data editor also has “types” and allows hex format and floating point notation. However, in the reference editor, all integers are unsigned. It allows a negative integer value to be set, but then displays it as unsigned. That is -1 can be set but it will display as 65535, and -32767 as 32769.

7.3.3 CanonicalEquation Format

The elements of the equation appear spaced out in the equation box in canonical format. The precise spacing and representation entered by the user is not preserved, only what could be deduced from an uploaded network from the PLC. You may have entered: 100012, however, 100012U is what appears, confirming to how the variable has been interpreted. Since the Equation Editor is treating the equation as one long ASCII string, a specific element never is identified to the user, not even when it is flagged as an element confusing to the parser.



7.3.4 Equation Complexity

The nodes of an equation network are:

the enable contact or short
the output register
up to 70 elements of an equation's expression
the 5 output coils

Operators, functions and parentheses take one node each. Variables of 0x, 1x, 3x, and 4x, and 1-bit (type B) and 16-bit constants take one node each. Floating Point and other 32-bit constants take two nodes each. The equation can not be longer than can be displayed in the space of the equation box even if it requires fewer nodes than the possible maximum.

Since the PLC must compile equation networks, assigning intermediate storage and intermediate operations, there is the possibility that a legal equation with 70 or fewer elements could still overflow the PLC's capacity for a network. In this unlikely event, (which the PLC flags as "run time stack overflow") an error dialog box will appear saying this or "intermediate requirements too great - simplify equation". The preference is to run the PLC's parser in the PC against an equation, only downloading when it has already passed logical parsing tests. The use of the PLC to check parsing is impossible in Offline mode.

If the PLC is in Online or Combined mode, the new network is downloaded if it passes initial parse tests, the PLC compiles the new network, at the same time, (if running), continuing to calculate using the old version. If, in spite of precautions, the network has errors, the PC gets the PLC's error code and element and the PLC continues calculating the outputs from the old equation. If there were no errors, it would simply execute the new equation. Although there is a discontinuity due to the difference between the old and new equations, there is no scan for which a calculation is not done.

7.4 Search and Trace

To Search or Trace, the Equation Box must be closed. An element inside the Equation Box can not be specifically indicated as the result of a Search or Trace operation.

The Search operation is different for Equation networks in two ways:

1) finding a searched for element in an Equation Network and

2) starting a search specifying an element in an Equation Network. An element found as the result of a SEARCH, if it occurs within an Equation Box, cannot be highlighted, Multiples are also not indicated.

There is no means to select a search element inside the Equation Box except by explicitly entering it in the Search dialog box. However many times the element appears in the Equation Network, it is only noted once and if the Search is continued, the element sought in other networks..

Trace is similarly limited by Equation networks. If a contact from another network is traced to a coil in an equation network, it is highlighted. The coils that could be traced to the Equation network are the five optional output coils. However, an internal 0x element in an equation box could not be Traced. The user would have to use Search instead. The enable contact is able to the Traced.

If an output coil of an Equation were Retraced, a message would indicate if it appeared in the Equation Box (but not how many times), and it could continue to be Retraced into other networks. Similarly a Retrace of a coil from another network finds an Equation Network in which the reference appears but the specific reference in the Equation Box is not indicated nor if it appeared multiple times. A request to *continue* Retracing would step to another network.

7.5 Equation Network Transfer to PLC and Program Files

If the PLC is offline and a program which includes equation networks is created or read in, before the PLC is put in online or combined mode, you must be sure that equation networks, in particular, are valid. There is possibility of invalid, saved equation networks because a program can be developed offline and be validated by the Modsoft version of the parser then fail when downloaded into the PLC's parser. If there are errors in any equation, then the equation networks containing them cannot be run in the PLC.

The standard ladder logic elements of the Equation Network can be edited in online or combined mode just as they are in regular ladder logic networks. They may be deleted and changed one at a time and the network is updated in the PLC while running. An Equation element could be complex enough that the PLC cannot just update it, compile it, and compute from it in one scan. However, it will not be noticeable to the user that it takes several scans, nor will the changeover from the old equation to the new one take more than one scan. In detail what happens is:

The equation element is opened to edit. While it is being edited, the PLC continues to calculate the Equation network under its old definition.

When the equation update is completed and recognized as valid, the new definition is downloaded to the PLC and the PLC "compiles" it. During this period, the output register is updated by the old version of the Equation Network. The output coils, similarly are set by the old calculations.

When the PLC is ready to use the new definition, it does so and the register and output coils' states are now set by the calculations under the new equation network definition. If there is an error in the new equation, it is discarded and the old one continues to be used.

Chapter 8
Reference and Symbol Editing

- Screen Orientation
- How to select References
- Reference Menu Topics

8.1 Reference And Symbol Editors

The Edit features that allow Reference values and Symbolic Data to be changed as part of the Network programming effort are a significant part of the PLC program. These editors are available from either Offline or Online Program development menu Utility function.

8.1.1 Utility

The Utility functions at this entry point (Reference Data Via Ladder Editor) provides you with the following functions:

Key Help Program Information Reference Data Symbol Table DOS

The entries for Key Help, Program Information and DOS have previously been defined and are not repeated here.

This Chapter documents all the Offline menu selected Reference and Symbol functions. When you are Online, these same menu selections have the additional function PlcOps. The PlcOps sub functions support PLC specific programming conditions and were described in the PLC Programming Chapter.

As you enter this area of functionality, you should review the Help keys available as illustrated.

```
Key Help for the Reference Data Editor
ALT-A = ASCII Mode                  ALT-M   = Matrix Mode
ALT-B = Binary Mode                 ALT-N   = Set the Range
ALT-D = Decimal Mode                ALT-S   = Save Template
ALT-E = Floating Point              ALT-W   = Change Window Size
ALT-L = Load Template               ALT-X   = Hexadecimal Mode
ALT-U = Long Decimal                ALT-C   = Clear Screen
ATL-I = Signed Integer              ALT-F1  = Symbol Table
CTRL-F1 = Program Information       CTRL-F7 = DOS
+     = Enter Next Reference        -       = Enter Previous Reference
Space = Enter Same Reference
ONLINE ONLY KEYS:
CTRL-F5 = Start                     CTRL-F6 = Stop

The <TAB> key toggles between an active editor and an active Menu Bar.
The cursor arrows are used to move the cursor across and down menus.
Typing the first letter of any menu item will move the cursor to it.
Pressing the related Function key will activate the menu selection.

For topic specific Help, pressing <Alt-H> will bring up information
on the selected menu item.
Navigate using <PgDn>/<PgUp> or <Esc> to quit
```

Figure 232 Reference Data Key Help

8.2 Reference Data Select

The quick key for the Reference Data Editor (RDE) is <**Alt-F2**>. By pressing the <**Alt-F2**> key, you can call the RDE from any editor without pulling down the Ladder Editor Utility menu. You can also move the cursor down from the Ladder screen.

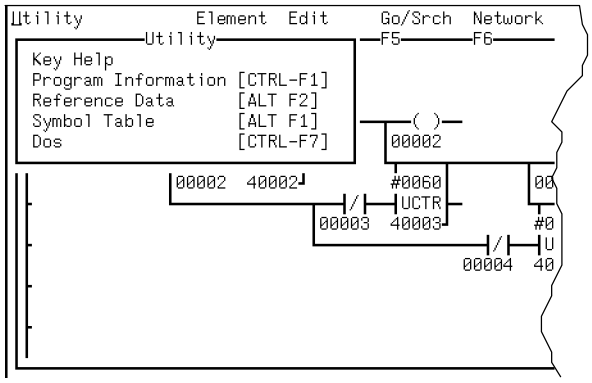


Figure 233 Reference Data Function Menu Selection

When you select it, the Reference Data Menu provides the following menu functions and places the cursor in the initial field of the reference data display area:

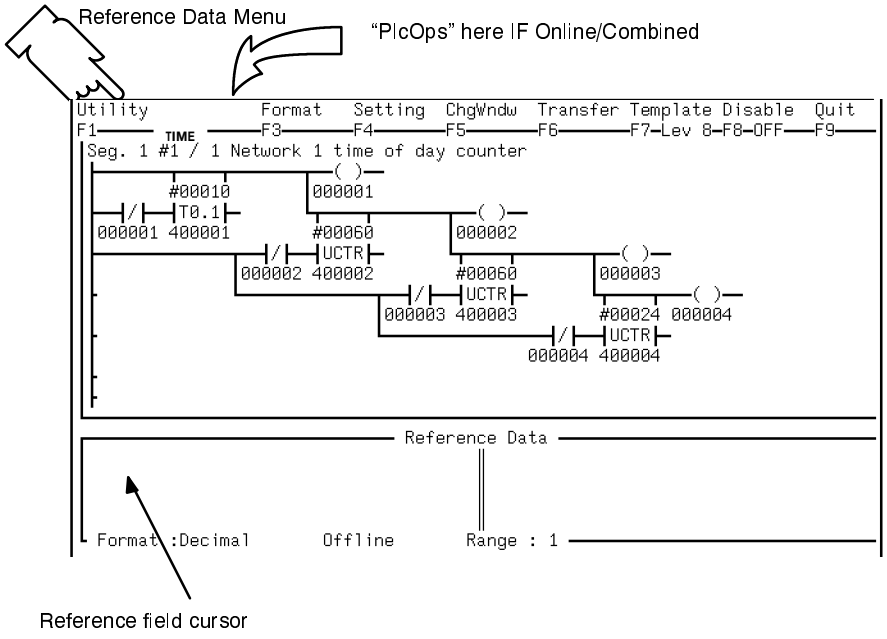


Figure 234 Reference Data Initial screen

8.3 Reference Data Editor

The Reference Data Menu combines the following menu entries:

Utility PLC Ops Format Setting ChgWndw Transfer Template Disable Quit

The PlcOps entry is only available in Online or Combined modes.

In the Online Mode, continuous monitoring of the values on the screen is provided until you decide to place the work in a file. The following illustrates an example of reading reference data from registers 400001 and 400002 and specifying a bit only in 400003. Variable data is updated by the PLC. Symbolic information comes from the loaded .RFS file is using Online Select Program or Combined mode. All entries to the Reference Data Editor can be saved to a file.

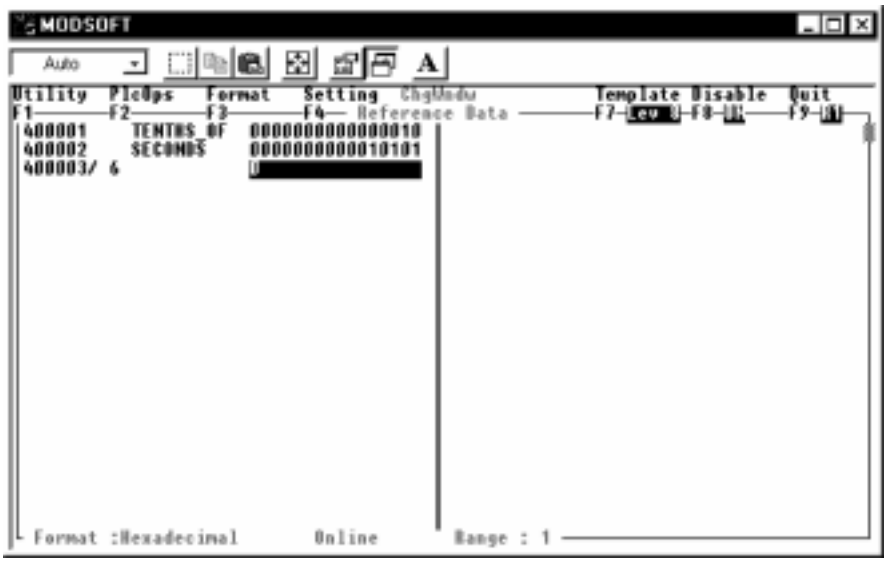


Figure 235 Example of Online PLC Dynamic Reference Data

This editor is used in Offline and Online modes to display, monitor, and transfer reference data to a configured controller or to a file, independent of a program editor. The work window of the editor consists of two columns, each containing the:

Reference
Reference Bit
Symbol,
representation of the value

relays and coils, have an Enable/Disable field.

The currently selected data format is displayed in the bottom line. The cursor is displayed as a block-cursor, and can be moved freely over the work window, with cursor keys. The RDE can hold up to 88 references.

☞ **Note** When you change project names offline, the current RDE template (temporary copy) is cleared. You will no longer see the RDE template carried from one project to another or from one online controller to another.

When leaving online, and going back online to the same controller, it is not possible to know if you are switching between controllers or back to the same one. Thus the RDE template is also cleared. If you wish to keep the RDE template, save it as a file and reload it when back on-line.

☞ **Note** Although you can enter a reference number beyond the configured range, there will be no associated data for the over range reference.

8.3.1 Mode Selection

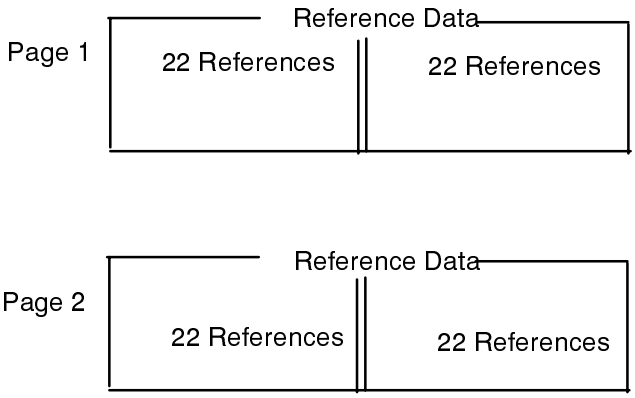
Choosing the Reference Data Edit (RDE) mode is independent from the rest of Modsoft , meaning, for example, that you may choose Load or Save Template even though the rest of Modsoft is “Online”.

The currently selected programming mode be it Online or Offline (including Combined) is displayed at the bottom of the RDE window as long as no pulldown has been displayed in which case the help line overwrites the mode line.

When the contents of the RDE window are modified in “Online” mode, Modsoft scans the screen and updates the polling list. If a communications error prevents the new references from updating on the screen, an error message appears, and the RDE mode changes to “Offline”.

8.3.1.1 Reference Data Editor Memory and Screen Layout

The RDE holds a total of 88 references broken into two pages of references with 44 in each page. Each page shows 2 columns of 22 references. You can page back and forth between the pages.



Reference Data Editor Screen

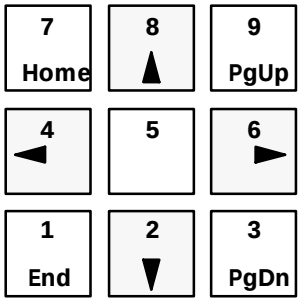
Each reference contains:

5 or 6-digit reference number: First 10 characters of symbol name
Up to 16-bit value: 2-digit special information
(Enable/Disable)

For text entry, the character cursor is displayed in the item field. The display refresh is suspended when there is any keyboard activity, such as moving the cursor with the arrow keys. As the user moves the cursor, the screen display does not update. Continuous updating resumes only when the keyboard is idle. The RDE window footer displays the format, Programming Mode and the range.

8.3.2 Moving Around The Screen

The right and left arrow keys move the cursor horizontally between the different reference fields and between the two columns of the current page. The cursor does not wrap around to the other side when an attempt is made to move it beyond the left or right border of the window.



Cursor Functions

The up and down arrow keys move the cursor vertically one row at a time. When the RDE is in “Offline” mode, moving past the bottom of page 1 scrolls the window up one row. Similarly, attempting to move past the top border scrolls the window down one row until the cursor is in the first row of page 1. The cursor does not wrap around to the top or bottom when an attempt is made to move beyond the outer page boundaries.

When the RDE is in “Online/Combined” mode, moving across the boundary between page 1 and page 2 snaps to the next page, meaning that you may only look at page 1 or page 2 but not a combination of both.

When in the RDE partial screen from the Ladder Logic Editor, access is limited to the first 5 rows in page 1. Access to page 2 is not allowed. A total of 10 references can be displayed in the partial screen.

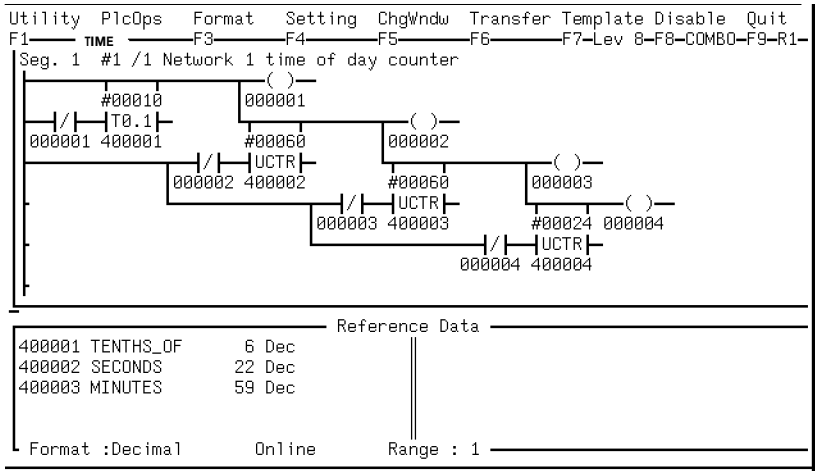


Figure 236 Example of Partial Reference Screen

Cursor cluster keys are used to move to the next page and put the cursor in the “home” position (upper left corner of reference data screen). If currently in page 1, <**PgDn** > goes to page 2. If currently in page 2, <**PgUp** > goes to page 1. The cursor does not wrap, so a <**PgDn** > in page 2 does not go to page 1. When in the RDE partial screen, <**PgDn** > does nothing. Page Up sends the cursor “home”.

The <**Home**> key moves the cursor to the upper left corner of page 1. The <**End**> key moves to the last row in the right column of page 2. To move to the “home” position in page 2, use <**Home**> and then <**PgDn** >. When in the RDE partial screen, <**End**> moves to the fifth row in the right column of page 1. Access to page 2 is not allowed.

8.3.3 Selecting References

The RDE screen is blank upon initial Modsoft start up. The RDE memory retains the layout of the window, if you go to another editor and back to the RDE later. However, if you leave Modsoft, the next session shows a blank RDE window.

When the cursor is on a blank row in a column, it can be positioned on one of three reference fields using the <**Left** ← **Arrow**> and <**Right** → **Arrow**> keys.

Reference Number	The leftmost (smallest, 5/6-digit) field.
Reference Symbol Name	The middle (10-character) field.
Reference Contents	The rightmost (largest, 16-bit) field.

Reference selection can only be done if the cursor is on the leftmost or middle field. Selection cannot be done in the rightmost blank field.

8.3.3.1 Single Selection

One reference at a time is placed on the screen, and you specify directly which reference number to display. In both numeric and symbolic cases, you enter the reference and then <↵ >. After a reference number is entered, the symbol (if any) and value appears in the other two fields. If you enter a a reference symbol name, the reference number and value appear. The <↵> key completes the selection and moves the cursor to the reference contents column. This allows you to enter a value without having to move the cursor after the selection.

Alternatively, the < ↑ > and < ↓ > keys complete the selection, then move the cursor up or down a row. This allows another reference to be selected without having to move the cursor back from the reference contents column.

In “Offline” mode, the entered reference is not checked with the configuration and is therefore always valid. This allows reference data files (Offline) to be created and values pre-set to be matched with a configuration at a later time.

8.3.3.2 Select by Reference Number

After placing the cursor on the reference number field, enter the numeric reference, starting with 0, 1, 3, or 4**;** for coil, input discrete, input register, or output (holding) register. Then enter the next four digits. For example (6 digit PLC), to select output register 400001 enter:

4 then 00001 then < ↵ > for 400001

If less than the full digit field is entered, the reference defaults to a coil (0x), and the reference number is front padded with zeroes. For example, enter:

456 then < ↵ > for 000456

You can use the **;** character to select an absolute memory reference, which assumes the next five characters are hexadecimal digits (i.e., 0-9, A-F).

This procedure works for both screen reference rows, an empty row within a column and a reference row already occupied. To change an existing reference, place the cursor on the existing reference number field and enter the new one. While entering a reference number, some special keys allow you to correct entry errors while in the field before the <↵ > key is used. They are:

< ← > moves left within the field.
< → > moves right within the field.

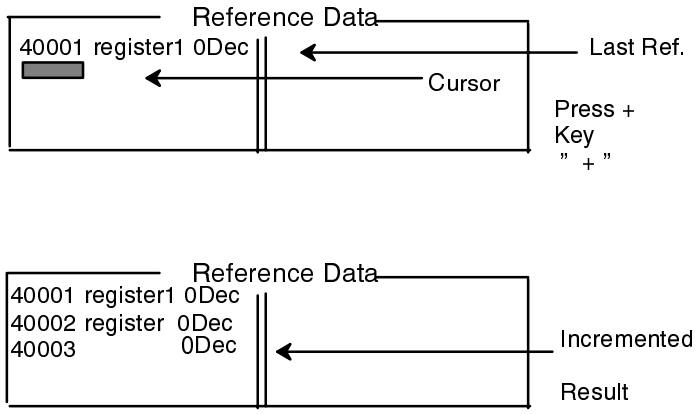
<**Ins**> toggles the ability to insert or overwrite within the field.
<**Esc**> cancels the entry returning it to its previous value.

After the < ↵ > key is used to complete the entry, the cursor moves to the reference field so a value can be entered or edited.

☞ **Note** The RDE cursor skips the symbol field. If you need to edit a symbol go to the Symbol Table Editor(Paragraph 8.10).

8.3.3.3 Select Next Reference

Each time a reference is added to the RDE window, the entry is remembered. The next number is easily added to the RDE window by placing the cursor on any blank reference row and using the <**+**> key. This feature works even if the blank reference location is not adjacent to the last added reference. For example**:**



Select Next Reference Number

This feature works for all reference types (0x, 1x, 3x, 4x, 6x). However, placing the cursor *on top of* an existing reference number (instead of at a blank location) performs the “Select Next Reference” feature using the reference number at the cursor position instead of the number stored in RDE memory.

8.3.3.4 Select Previous Reference

The same rules apply as for “Select Next Reference” (above), but the <**-**> key is used to subtract one from the reference number and put the resulting reference to the RDE window. This feature works for all reference types (0x, 1x, 3x, 4x, 6x).

8.3.3.5 Select Same Reference

The same rules apply as for “Select Next Reference” (above), but the <**Space**> bar is used to repeat the same reference number and put the resulting reference to the RDE window. This feature makes it easy to view the same reference in different formats (decimal, hexadecimal, and binary,) by later using the Setting pull-down. This feature works for all reference types (0x, 1x, 3x, 4x, 6x).

8.3.3.6 Single Deletion

Place the cursor on any field within the reference row and press <**Del**> to remove that reference from the RDE window. You can delete more than one at a time with Clear Screen <**Alt C**>. You can delete all references in the RDE partial screen, using the <**Del**> key.

☞ **Note** Even though the reference no longer appears in the RDE window, it still exists in the controller and in the Offline file.

8.3.4 Bit Descriptors

The Reference Data Editor display is either at the bottom of the ladder screen or in the expanded display. To accommodate bit descriptors, the reference data field width is increased to 9 characters. This allows you to enter the reference and bit number in the field. To facilitate this, all fields on the display (after the reference number field) are shifted right 3 places, with a minor revision of the layout.

For example:

400003/04

If this is a legal reference with bit number it will be replaced by the following:

400003/04 Bit_symbol> 1 or 0 (for example).

If no bit descriptor exists, the symbol is blank, but the register bit will be displayed and is available to edit.

The register bit may also be called from its symbol just as any other reference.

8.4 Format

This function is provided to show values in different formats. The Pulldown Menu items are displayed as:

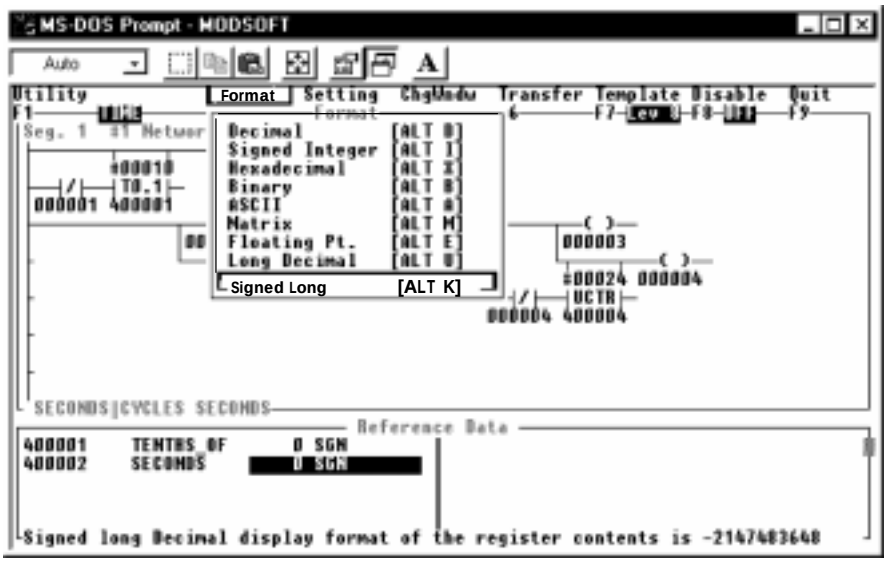


Figure 237 Format Menu Select Options

In all modes but matrix, values can be changed by simply typing the new value in the appropriate value field. References can be enabled or disabled by typing “E”/ “e” or “D”/“d”, in the Enable/Disable column. References can be forced on or off by typing ”1” or ”0” respectively in the value field. The selected mode is valid for further reference specification.

Single references can be displayed in binary, hexadecimal, decimal, Floating Point, ASCII or matrix mode (0X, 1X only) by selecting the appropriate menu item for the reference at the actual cursor position. The format for the reference selected is located to the right of the reference value. The default mode setting is decimal.

8.4.1 Editing Values

Once a reference is selected in the RDE window, its reference value can be changed. Using the arrow keys, place the cursor on any reference value field and enter a new value, followed by < ↲ >. If you press the return key without making a change you move the cursor to the next row. Alternatively, the < ↑ > and < ↓ > keys also complete the editing, then move the cursor up

a row or down a row. This allows another value to be changed without having to move the cursor back from the reference number column.

The display format determines which characters are allowed:

Decimal:	0 - 9
Signed Integer	+number or -number
Binary:	0 and 1
ASCII:	(See the 984 Systems Manual GM-0984-SYS or any standard ASCII code chart).
Hexadecimal:	0 - 9, A - F or a - f
Matrix:	none, this format is read only
Floating Point:	whole number with decimal point and remainder. (two registers are used to store the internal number). the dynamic range of this format is +/- 1.175494E-38 to 3.402823E+38 with a 7 to 8 digit accuracy.
Long Decimal:	Long Unsigned Decimal Integer (0 to 4294967294) with 2 registers required.
Signed Long	Long signed decimal integer

In Decimal format, entering a value greater than 65535 rounds the entry back to 65535. The format label (Dec, Hex, etc.) disappears during editing and returns when editing is complete.

Note When in "`Online/Combo`" mode, with the controller stopped, it may take up to four seconds for the format label to appear.

Tip: Decimal display does not show leading zeros.Be careful when interpreting the results of EMTH operations which devote a register to the decimal part of the answer, i.e. the result must always be interpreted as having 4 digits; missing leading digits are zero:

blank3 00 is read as **0**3 00

Matrix mode is a special feature to display 16 0XXXX or 1XXXX references. The values are displayed in 16-bit clusters, always starting at a 16-bit boundary. Within this mode it is not possible to edit values of these references or to enable or disable them.

8.4.1.1 Coils and Input Discretes (0X, 1X)

These are edited only in decimal format. The "ON" or "OFF" state is entered in the reference contents field as a 1 or 0. The on/off state cannot be changed if the discrete is "enabled". Us-

ing the < ↵ > key when editing these two fields moves the cursor to the other field. This is useful, for example, to disable a coil or force it on with a minimum number of keystrokes.

8.4.1.2 3X 4X 6X References

There are no special editing features for these references.

8.4.1.3 Absolute Memory (Page 0 - F)

When in "Offline" mode, the contents field is blank and cannot be edited. You can only display the value at a specified location.

The first digit designates the page (0-F) and the next four are the address in Hexadecimal notation. (Case A-F is not sensitive, so 'a' is the same as 'A'.) For example, entering:

;012af then < ↵ > is the same as ;012AF

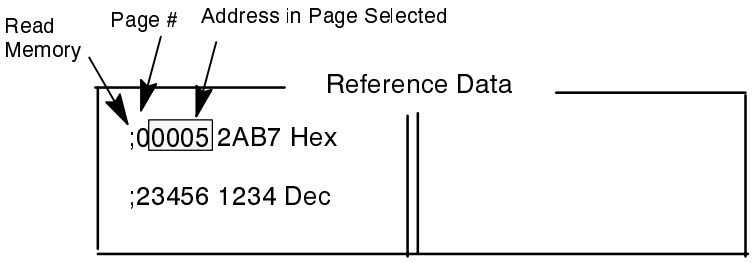
Absolute addresses do not pad with leading zeros after the **;** symbol.

;32 then < ↵ > for ;3 0002 i.e., page 3 address 2.

6x entries are for extended memory controllers 984B and 785E or L only, and require you to designate a file number in the second digit of the 6x reference i.e.:

6**0**0001 is 60001 file 1 and
6**9**0001 is 60001 file 10

the user must start the number with a '**;**'.
For example:



Reference To Memory Location

No editing of the contents of page 0 through F are allowed. Absolute data values that are displayed are not stored in Offline file and can not be accessed with "`Transfer`" menu.

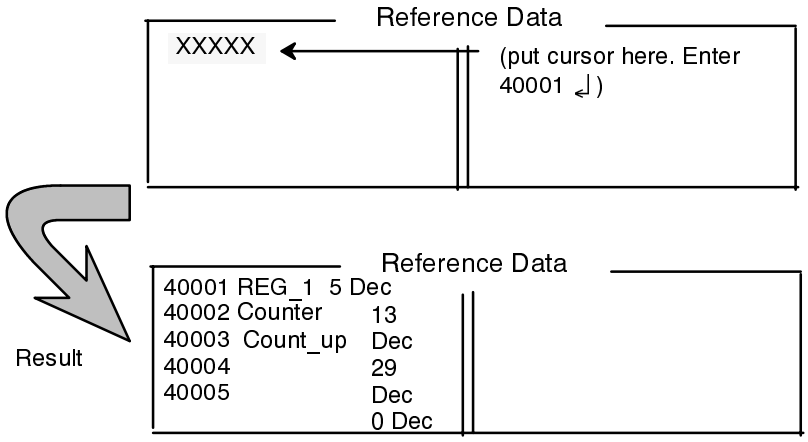
8.4.1.4 Quantum Memory Reference

A memory reference feature is available in Quantum to allow read and write of memory locations in pages A0 to FF of memory. This requires special treatment of data entry in the address field. You may enter as usual `;xxxxx` to read any memory in pages 0 to F. *In addition a new mode of entering the address field will allow axxxxx to fxxxxx as special cases of the memory addresses at A0 - FF.* If you enter such an address in other than a Quantum controller, an error will be flashed.

8.5 Setting

8.5.1 Range

The Range selection allows any single selection operation to add many sequential reference numbers to the RDE window . The current range displayed in the RDE window footer, is defaulted to 1, and stays in effect for the duration of the edit session. Any range number from 1 to 88 is allowed. For example, if range is set to 5:



Setting Reference Ranges

This feature works when entering reference numbers only. Entering a reference symbol name always displays one reference, no matter what the range is set for.

☞ **Note** In the RDE partial screen, any range larger than 10 is interpreted as 10. The 10 references added to the screen wrap at the bottom edge of the RDE partial screen and continue down the right column. The hidden references on the rest of page 1 and all of page 2 are not affected.

☞ **Note** In the RDE full screen, the references continue to be added in the left column, including page 2, and then move to the top of the right column of page 1. Once page 2 is filled (to the bottom of the right column), any remaining ranges are not displayed.

You can also select the `Window Size` function or the `Clear Screen` function from the Setting pulldown.

8.5.2 Clear Screen

The Clear Screen selection (Quick Key <**Alt C**>) Clears any displayed reference data from the current display.

8.5.3 Simple ASCII

A window is opened to accommodate this function. You are asked for the start address of the Simple ASCII register. You can enter up to 64 characters of text. It makes no difference whether these registers are later used for Simple ASCII input or output. There is no check against the register you defined for Simple ASCII in the configuration. You can edit any 4XXXX register, that may be copied to Simple ASCII registers later.

☞ **Note** When using Modsoft to enter simple ASCII data, pressing the <**ENTER** > key terminates the field. If you need to explicitly code a Carriage Return / Line Feed, you must go to the Reference Data Editor and add the CR/LF value in Hex ,i.e. 0D/0A.

8.6 ChangeWindow

This RDE Menu item allows you to toggle between the limited display area found on the bottom of a Ladder display, to a full page reference display.

You may view more than the partial reference screen by going to the Setting function and selecting “Change Window Size”. You can toggle from full reference to the partial screen by selecting the “Change Window Size” **< Alt W>** function.

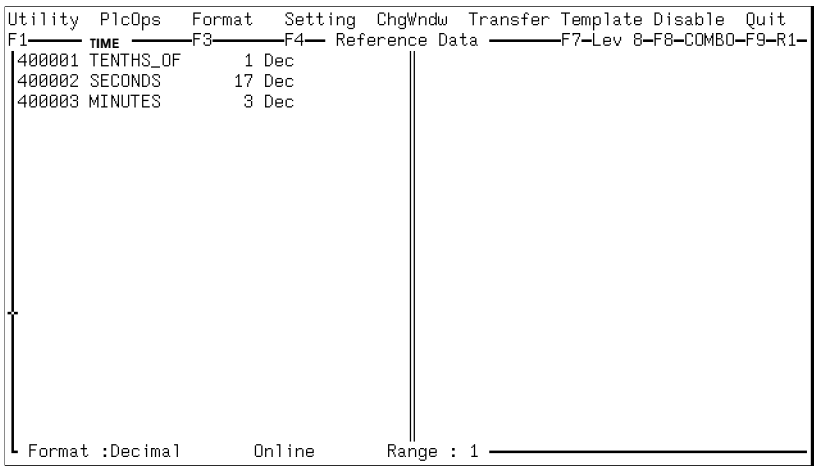


Figure 238 Full Screen Reference Window

8.7 Transfer

There is an ***Offline*** import and export facility to load as well as to store a file in standard ASCII text format. An export file can be edited using a text editor and imported into the Reference Data editor, provided the format of the file is maintained. You are prompted for a “Filename,” which allows selecting from available media by inputting the full path name.

8.7.1 Import

This feature reads an RDE exported file. You specify the file name. The same rules apply for adding new references to the Offline file via the RDE window (or loading a template). Comments can be included in an Import RDE if prefaced by a / character.

Import will look up each reference number read in the Offline file. If a number is present, take the value from the import file and write it in the existing reference record. If there is no number, add a new reference record including the number and value to the Offline file. Thus, import has an accumulative effect. Several files may be imported so the Offline file reflects the combined set of all the references in all the imported files.

8.7.1.1 Export

Reference numbers and contents are written to a flat ASCII disk file in the DOS directory. You specify the path and filename. Only those records found in the Offline file are exported. Below is a reference export with its format where each line of the exported file starts with the reference number.

Exporting to external data base editors like Microsoft Access or Borland Dbase allows viewing Modsoft data in the .RFS and .PCM files.

If any external edit is made and file is restored to Modsoft you must assure that Modsoft can still use the file. The .RFS and .PCM file structure may not be fully compatible with Borland Dbase.

In particular adding a new record to .PCM files can cause difficulties in Modsoft, regarding operations that use the long comment Tag.

If a problem is encountered it may be possible to recover by accessing all the fields in all the offending records and checking that at least one character (even a Space) is present in the field. If this precess does not work you will have to go back to the external database to make corrections or deletions!

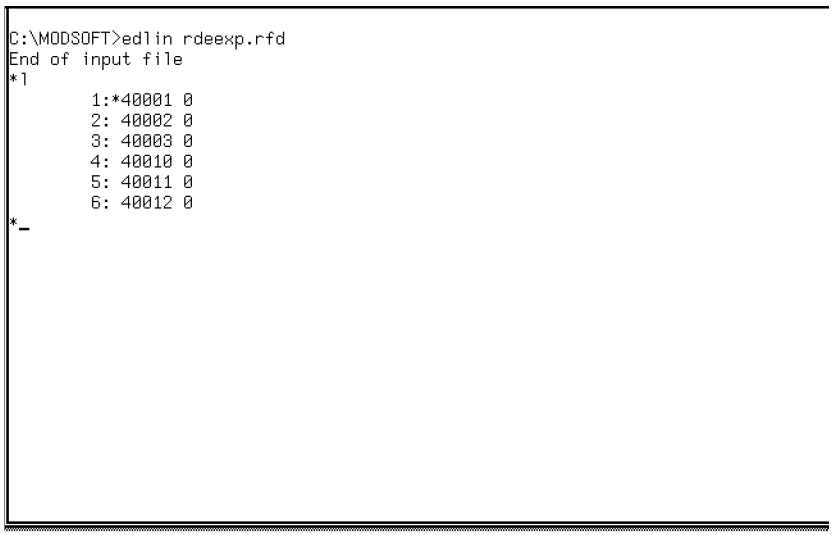


Figure 239 Export File Example

☞ **Note** You should use this feature on a Read-Only basis for documentation viewing. If you use an external editor to alter records, the file structure may become incompatible to Modsoft.

8.8 Template

8.8.1 Save Template[Alt S]

The RDE template feature allows you to take a “snapshot” of the RDE screen layout (all 88 references) including the formats. The reference values are not stored in the template. The selection prompts for an 8-character file name and adds the .RFW extension.

The <**?**> key displays a selection list of already saved templates in the “Reference Data Path” displayed by the Program File Settings function. Use the < ↓ > , < ↑ >, <**Home**> and <**End**> keys while inside a selection list to select the file name, in addition to typing the first letter of the file name.

The RDE screen layout is written to the template file and can be loaded later using the “Template” menu “Load Template” entry. The selection list works here as well.
This feature is helpful to save and later load combinations of references that relate to online sub-systems for online troubleshooting.

A quick method to clear the entire RDE window memory is to use the Clear Screen command under Setting

When Saving there is a default path to be the programs path for the RDE template filename The filename display request is expanded to allow entry of the full path, with the default path being the currently selected programs directory. In addition, the template name is defaulted to the project name (.env file name). This prompt appears as follows, for example:

Filename: \modsoft\programs\project1 .RFW

(The .RFW extension is non-changeable).

8.8.2 Load Template[Alt L]

If the “SaveTemplate”function has been called previously, this command restores the previously saved reference formats in the reference data window. The values and the status information are updated after the template is displayed. The contents of the temporary file stays unchanged. Having once called the “Save Template” function, “Load Template” can be executed, even if the reference data editor had been exited and accessed again. A list of previously ‘Saved’ templates can be shown by pressing <**?** > after selecting the load or save template function.

8.9 Disable

Within the Reference Data Editor, the following commands are provided:

- Enable All
  Enables references previously set to “D” (disabled).

- List Disabled (Online Only)
  You can List Disabled Coils only in Online mode

8.10 Symbol Table

The "Symbol Table" under the Utility menu pulldown provides an editor to create, change and delete symbolic names and comments for the references in the controller program. This text-based editor creates and maintains a table of reference symbols. For each valid reference, you can assign a symbol of 32 characters, and a Descriptor of 196 characters.

The first symbolic character string can consist of letters, digits or _. This first symbol must begin with a letter or _ and must be unique within the table (upper or lower case letters are acceptable and are treated the same).

Note Symbol Table is not available in Online - Direct to PLC. It is available in Online → Select Program Offline and Combined

Expert When running with "iec_standard" = zero in the User Preference File. Symbols can be any characters, including spaces, as long as at least one character is a letter. However it is recommended that Symbols start with a letter, digit, or underscore, thus preserving a format that significantly enhances your ability to move Modsoft symbols to other software applications.

This is a sample table to familiarize you with the data fields as explained above.

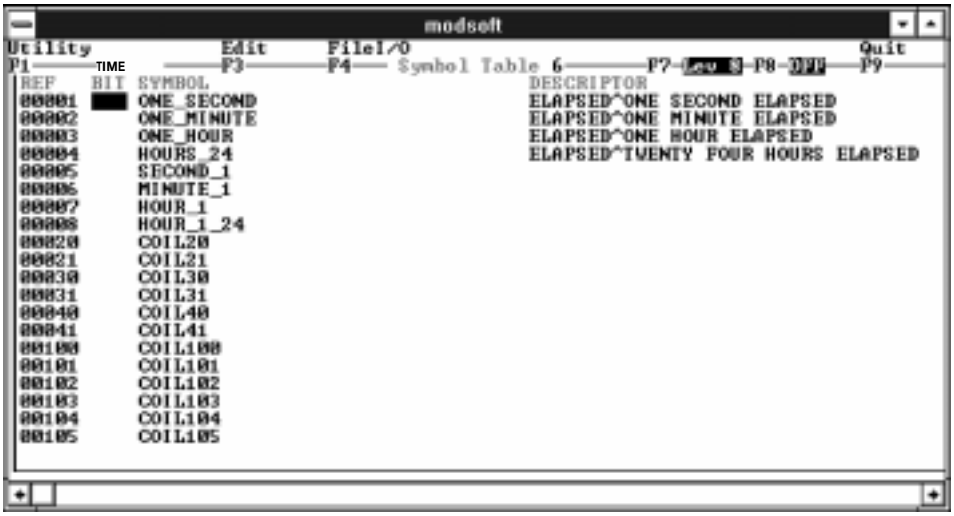


Figure 240 Example Of Symbol Table

The references are not checked against the current controller configuration, because the same reference table can be used for any number of programs and controller types.

During an edit, when the cursor is in a specified field, the <**Del**> key deletes characters and the <**Ins**> key allows character or string insertion. If the <**Ins**> key is not pressed, the field is in “Overwrite” mode.

> **Note:** This editor is character based. You can move the cursor and insert or overtype characters. Deleting a line of description using the backspace or delete key does not close up the space between lines. An empty line must be retyped if you need to close up the text.

When entering the Symbol Table Editor, the Symbols Menu appears, and a titled window is opened, which holds the first 20 element lines of the symbol table. Each element takes up one line. The Edit Symbols Menu contains the following functions:

Utility PLC Ops	Edit	Quit(If Online / Combined)
Utility Edit	FileI/O	Quit(If Offline)

Figure 241 Example of Reference Insert with Symbol and Descriptor Edit

Once you have initially created a Reference Symbol and/or Descriptor from the Symbol Table editor selection (of the `Utility` menu) you can update the text by reentering the descriptor field and typing any key or by using the Ladder Diagram `Ref Zoom` pulldown.

If you use all four fields only the first one is displayed in the Symbol Table.

> ☞ **Note** If you place carots (**^**) in the descriptor, the lister places line breaks at these points.

You can define symbols for constants. The >#XXXX< (X:any digit) reference numbers are constants.

The elements are maintained in reference order unless the last sort requested was symbolic, in which case they are kept in symbol order. Your edit sequence is kept until a “Sort by” command is issued. You can choose to sort by reference number or alphabetically by ASCII order. If sorted, the window is redrawn in the new order, and the next time the editor is entered, this sort order is used.

The cursor does not cover the complete line but only one entry field within that line. The cursor can be positioned in either the reference field, the symbol fields or the comment field.

When the cursor comes to the top or bottom of the window, the elements are scrolled. The <**PgDn**> key shows the next 20 elements, the <**PgUp**> key the previous 20. <**Home**> moves the cursor to the first line of the file, <**End**> moves it to the last element.

To modify the assignment between a certain Reference Number, Symbol, and Descriptor, you have to move the cursor to the appropriate field, and either press the <**Ins**> key or immediately type in the first character of the field. A small character cursor appears in this field and allows you to change it. The <**Enter**> key accepts the change and <**Esc**> restores the previous state. If a reference number or a symbol has been modified, the system checks to assure it is unique within the table. Modifying the last line causes a new element to be appended at the end of the symbol table.

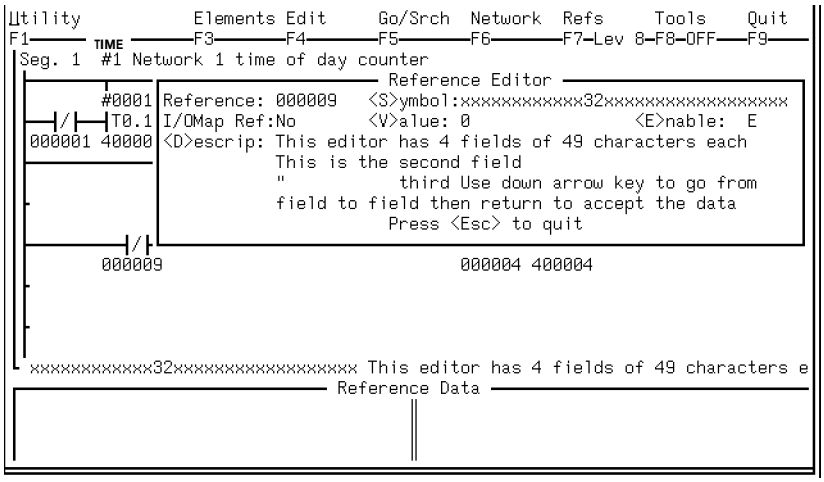


Figure 242 Descriptor Edit from Refs Edit Menu

8.10.1 Plus and Minus Key Editing

During a Symbol Table edit session you may use the “+” or “-” keys for quick key actions with the following conditions:

In the Reference Number field

(+) on the top row of the table results in an error message.
(+) on any other row and Modsoft tries to insert a line with reference number incremented one digit higher than the value above the insert point. If the reference number exists with the bit of 0, an error message (Reference Number already exists XXXXXX). If conditions allow, the insert, a blank lline is displayed with the edit cursor in the REF field.

(-) With this key applied to any row with the cursor in the reference number, Modsoft attempts to make the previous reference of that type and insert it in the correct row.

If you press the - key on a row without a reference number or a new reference exists at bit 0 you will get an error message. Given the proper conditions this key inserts a new reference in the row with BIT 0 blank.

In the BIT number field

(+) On the row below a legal reference number, Modsoft tries to insert a row with bit # incremented. If the reference and bit exist an error is displayed.

(-) On any row Modsoft tries to insert a row at the current location with the BIT number decremented. If the reference and bit already exist an error message is displayed

☞ **Note** BIT references can not be less than 0 or greater than 15.

8.10.2 Bit Descriptor Format

In the Reference Symbol Editor, there is an additional field immediately following the register reference number for bit number. If a bit number is entered, the symbol and description field apply only to that particular bit of that particular register (either 3X or 4X). An example:

300015/03	Symbol_for_30015_03_xxxxxxxxxxx	Up to 196 characters of descriptive text for this bit.

8.10.3 Editor Commands

When selected, the commands and Quick Keys in the pulldown sub-menu are available. You should also review the help screen <**Alt H**>.

Expert After creating a descriptor using Insert (for example), you can edit that field by putting the cursor on it and keying in an < ***** >. This action causes the descriptor to display on the same screen in an editing window. As illustrated below the insert is sorted for you

Figure 243 Symbol Table Edit Commands

8.10.3.1 Insert

This function inserts an entirely new Reference, unique Symbol, and Descriptor fields into the table of pre-existing references and comments. The function inserts a new line before the current position where the user can enter the new table element. The actual cursor line and all following lines are shifted down one row.

If you terminate your input after the reference number or first symbol with the <**Esc** > key, the command is ignored and the new line is erased on the screen. Otherwise, the new element is written into the reference symbol table.

The system checks whether the reference number or the reference symbol are defined already. If you have repeated a prior entry Modsoft reports the condition.

Figure 244 System Message for Error Input

If the cursor is on an empty field, you can increment or decrement the previously edited reference number by pressing the <**+**> or <**-**> key. You then key in the remaining data from the keyboard. The new reference number must be within the previously defined (configured) range. The shorthand for the insert function is <**Alt I** >.

8.10.3.2 Delete

This function deletes the reference number, unique symbol, and short / long comment fields when the cursor is on a reference row. After selecting “Delete”, a Delete marker is displayed in a color different than the frame background. The marked characters blink to emphasize the operation. Use the following cursor keys can be used to delete elements in the table:

<**Home**>, <**End**>, <**PgUp**>, <**PgDn**>, < ↑ >, < ↓ >, <**ENTER/Return** (↵) >, <**Esc**> leaves the table unchanged. The Delete shorthand is <**Alt F4** >.

In the next figure, the top three lines have been marked for a Delete operation. Pressing the <**ENTER/Return** > key executes the command, removing the symbols and comments from the editor.

Figure 245 Example Of References MarkedFor Delete

8.10.3.3 Copy Descriptor

This function only copies the descriptor field. After selecting “Copy”, a copy frame, shown in a special delete color, can be defined using the cursor keys as described under “Delete”. <**Enter**> copies all comments within the frame into the paste buffer, <**Esc**> quits the command. If the paste buffer is not empty, it is overwritten. The shorthand key is <**Alt F3**>

8.10.3.4 Paste Descriptor

The Paste function is only allowed if the paste buffer is filled by a previous Copy or Delete. All comment fields beginning with the current one are overwritten with the contents of the paste buffer. If the rest of the symbol table is shorter than the paste buffer, the superfluous comments are ignored. The shorthand key is < **Alt F5** >.

☞ **Note** The Copy, Delete and Paste Commands all use the SAME Buffer

8.10.3.5 Offset

Using this function, you can transform a range of operands to other addresses. You are prompted for the first and last reference number and the offset. None of the reference numbers can be a constant (#XXXX). The offset can also be negative.

Figure 246 Example Of Reference Offset

If a transformed reference number already exists, an error message appears and the command is ignored. The number of replacements is reported when ”Offset” has been executed. When you have the cursor in the menu area, the edit screen is updated with the new value (old value + 100 in the example) as illustrated.

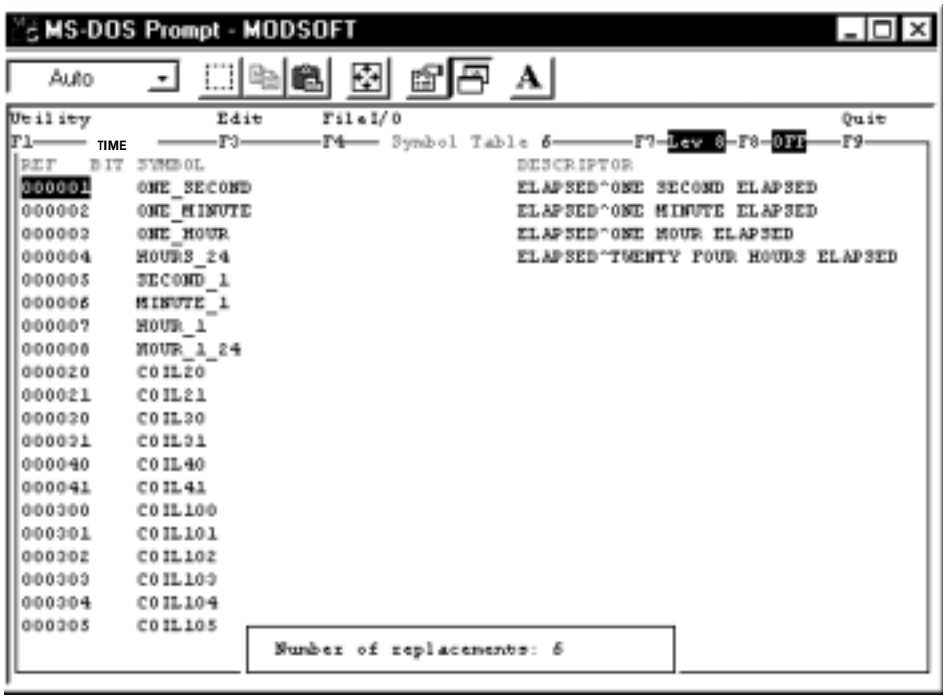


Figure 247 Offset Data Updated

8.10.3.6 Search

You can search the table for a specific reference number or a first symbol. If the searched element exists within the table, the cursor is positioned on it, otherwise an “element not found” message is issued. Constants can also be searched for.

8.10.3.7 Sort By

This function allows a sort of the Reference Symbol Table either by Reference Number or alphabetically by Symbol order. The heading of the table is always displayed afterwards. Your initial default is to arrange the table by reference number. When you have made a selection from the list, the sort by selected stays in effect until you change it.

☞ **Note** Sorted order is not saved when you exit Modsoft.

8.10.4 File I/O (Offline Only)

Using Import and Export file operations enables Symbols and Commands to be edited in an ASCII format.

- Selecting "Import" you can insert reference symbol elements into the table before the current cursor position. The next Figure shows you how to position the cursor for the addition of a reference from an external file, disk drive A: in this example. The result of the operation is also illustrated.

  The Import file is processed line by line in the same format as an export file (see next Paragraph). Existing reference number entries are overwritten. If a symbol exists already, the line is ignored. A warning is issued in either case.

Importing a Table Of Symbols text file has an option to not overwrite existing records without verification. The initial selection default is to `No`, however if you change to `Yes`, that remains the default until changed again. If you select `All`, or `None`, the remainder of the file will be treated in that way, without further intervention.

☞ **Note** If you press the `Esc` key to terminate operation of this function, the Import is quit, but all changes made up to this point are saved, not canceled.

If you create an independent file using an ASCII editor as in the Import case you must maintain a space between fields. Comment lines can be included if prefaced by a / character. These are ignored on Import.

- Selecting "Export" from the Pulldown Menu prompts the user for a file name. Existing files are overwritten if the user permits it. An Exported Symbol Table file allows the user to view the list for changes. The edited file can then be "imported" into the table. The next figure is an example of Exporting the Symbol table to disk A: file name ***export.***

☞ **Note** You should use this feature on a Read-Only basis for documentation viewing. If you use an external editor to alter records, the file structure may become incompatible to Modsoft.

Figure 248 Example of Adding Reference From External File

REFERENCE			UNIQUE		COMMENT
5 or 6 characters No Spaces	B|T	space	32 Characters	space	Reference Comment up to 196 Characters

Export File Format

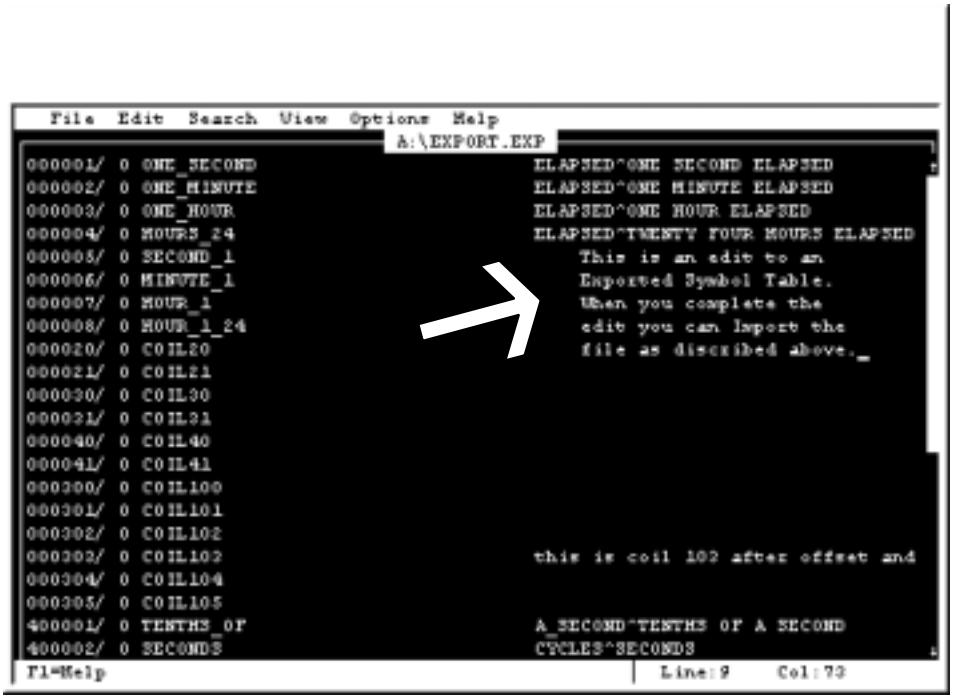


Figure 249 Example of Symbol Table Export

Chapter 9
Sequential Function Chart Programming

□ This Chapter describes the Modsoft implementation of the Sequential Function Chart (SFC) programming technique.

9.1 Sequential Function Chart (SFC)

The Sequential Function Chart (SFC) is a graphic language which is used in conjunction with Ladder Logic for programmable logic controller programming. The SFC language elements (steps and transitions) provide a means of organizing the control flow of a controller program. These elements are connected by directed links. The SFC Editor enables you to create, modify and delete steps, transitions and references, as well as create links between these elements (links represent the sequence control).

A step corresponds to a situation in which the behavior of the system with respect to its inputs and outputs follows the rules defined by the associated networks of this step. A step is either active or inactive at any given moment.

The state of the system is completely described by the active steps. Only the ladder logic in active steps is executed. A step remains active until its following transition is true, once true control flows out of the step to the step following the transition.

You can move around in SFC by using the cursor keys, the <**PgUp**> and <**PgDn**> keys, and the <**Home**> and <**End**> keys. As the virtual SFC screen is bigger than the actual screen, the chart is scrolled whenever the cursor comes to the top and bottom of the actual screen. Horizontal wrapping for the cursor is provided.

9.1.1 SFC Structure

The following series of visuals will help help you interpret the previous rules. The use of steps, transitions and links can be combined in only a given number of ways:

9.1.1.1 SFC

To use SFC, you must indicate the intent by turning **on** the SFC flag in the Configuration Overview menu and in the UPF File. Sequential Function Chart is initiated via the Segment Status display menu Editors pulldown which lists SFC as a selection. The quick key is [Alt S]. The Initial Display when the Time example program set is selected is:

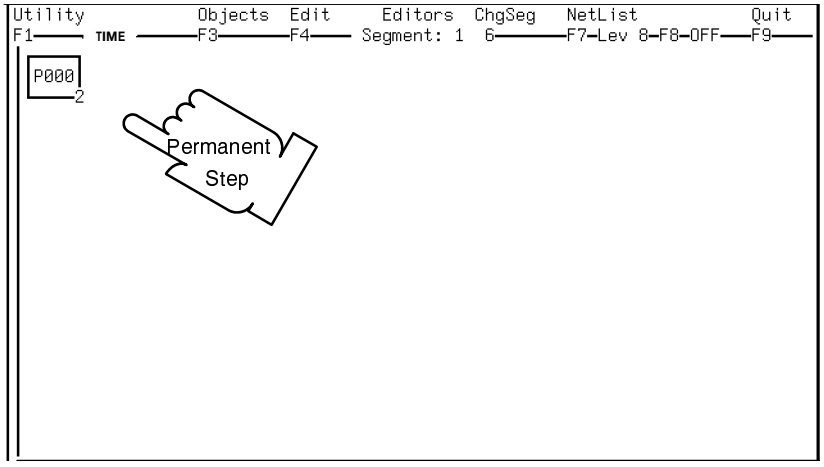
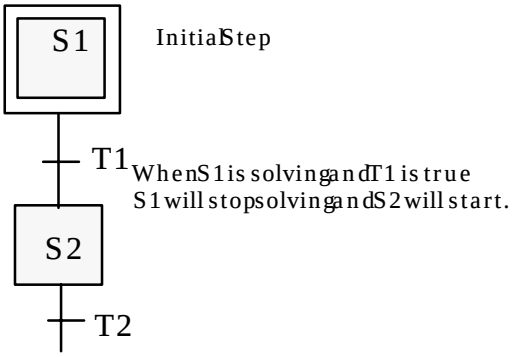


Figure 250 SFC Initialization Screen

The following sequence of instructions illustrate the rules and techniques needed by SFC Programmers.

9.1.1.2 Sequential Link

One step with its transition follows the preceding step.

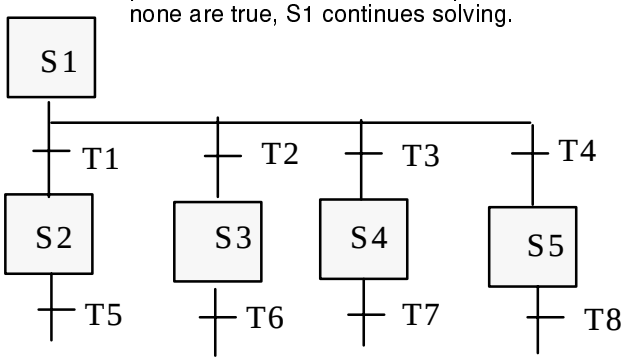


SFC Sequential

9.1.1.3 Selective Open

Following a step, there are a number of transitions. The leftmost transition that is true determines the solve path.

When S1 is solving, each of the T1-T4 transitions will be examined left to right. The first that is true determines the execution path. If T1 is true, S1 will stop and S2 will start solving logic.If none are true, S1 continues solving.

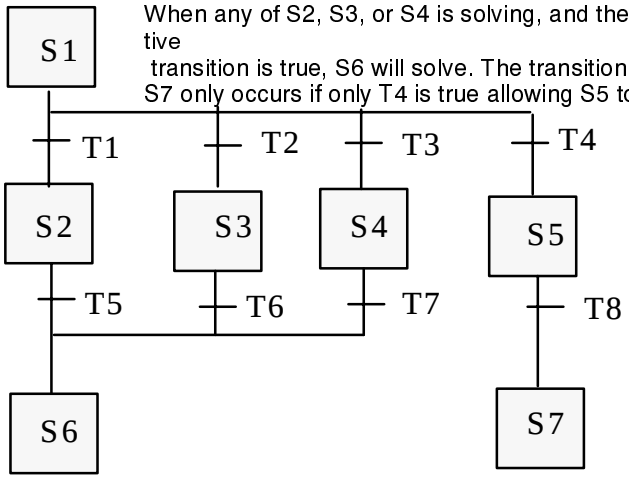


SFC Selective

9.1.1.4 Selective Close

When a selective open has occurred, a selective close is used to bring all the potential paths back to a single line of execution.

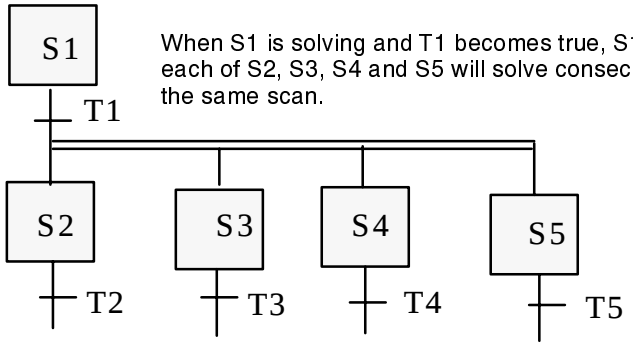
When any of S2, S3, or S4 is solving, and their respective transition is true, S6 will solve. The transition from S5 to S7 only occurs if only T4 is true allowing S5 to solve.



SFC Selective Close

9.1.1.5 Parallel Open

Following a transition is a number of steps. Each step will start solving when the transition is true.

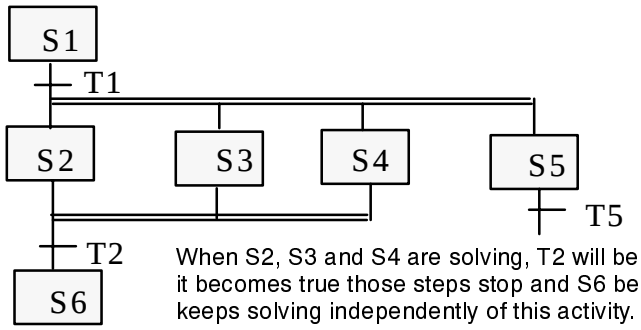


When S1 is solving and T1 becomes true, S1 stops and each of S2, S3, S4 and S5 will solve consecutively on the same scan.

SFC Parallel

9.1.1.6 Parallel Close

One transition may follow parallel steps. This transition will be examined only when ALL its preceding steps are solving.

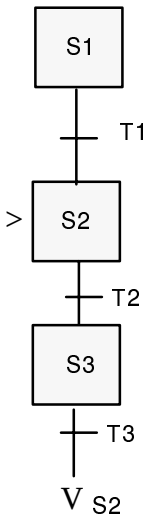


When S2, S3 and S4 are solving, T2 will be examined. When it becomes true those steps stop and S6 begins solving. S5 keeps solving independently of this activity.

SFC Parallel Close

9.1.1.7 Reference Link

To complete the SFC cycle, or when a jump to another section of a chart is necessary, a reference link can be placed after a transition.



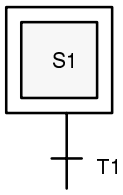
When S3 is solving and T3 is true,

S3 will stop solving and S2 will solve.

SFC Reference Link

9.1.1.8 Initial Step

One and only one step must be marked as the initial step and is started on the first scan of the program.

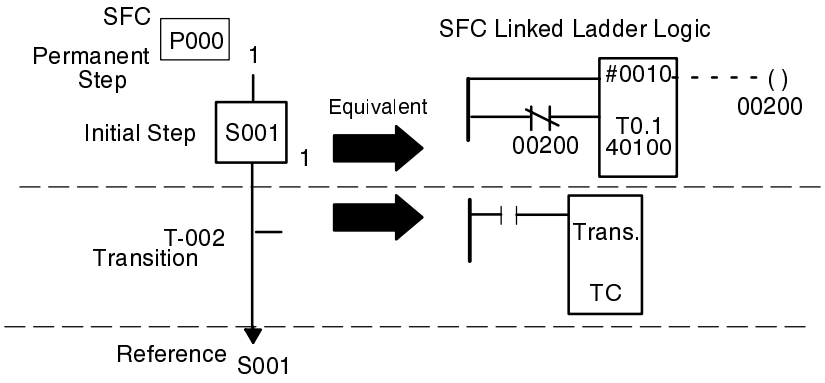


Initial steps are indicated by a double lined box.

SFC Initial step

The next drawing is presented to relate the concept of SFC program implementation. To a control programmer who uses a flow chart as a design tool and codes ladder logic as a separate entity, the binding of these functions is something new. However, if you observe programming languages at large, and consider SFC as a high order language, its structure becomes more apparent. To further this point, the initial step SFC, and a Network are included. These elements are the equivalent of a high order language source code statement. A second source line is represented by the Transition, Zoom and Transition logic block.

It is important to understand that the code does not have to be generated all at once. In fact, the prime use of SFC is to control logic execution using the familiar flow chart structure. It will be common to generate the flow first, then “append” the logic. When that sequence is complete you have programmed the source code.



CONTROLLER' 'READY" CONVERTED LOGIC (FLAT)

		DESCRIPTION
#00001	ADDITIONAL NETWORK	N.O. contact to a SKPC block
#00002	ADDITIONAL NETWORK	ADD block MBIT block and a coil
#00003	ADDITIONAL NETWORK	2 SENS blocks and 2 SKPC blocks
#00004	INITIAL STEP LADDER	As above
#00005	ADDITIONAL NETWORK	ADD block
#00006	ADDITIONAL NETWORK	1 SENS block 2 SKPC blocks
#00007	ADDITIONAL NETWORK	MBIT block
#00008	TRANSITION	1 SENS block 2 SKPC blocks
#00009	ADDITIONAL NETWORK	MBIT block
#00010	ADDITIONAL NETWORK	1 SENS block 2 SKPC blocks
#00011	ADDITIONAL NETWORK	MBIT block
#00012	ADDITIONAL NETWORK	Permanent Step Logic
#00013	ADDITIONAL NETWORK	
	REFERENCE	
	P000	

Figure 251 SFC ConceptDrawing

☞ **Note** The above is provided for concept only, and is only valid for the simplistic SFC illustrated.

Just as a high order computer language undergoes an expansion when compiled, so does the SFC code. The SFC equivalent to the compile process is called “CONVERSION” and the resultant code, instead of object, has been called “FLAT LOGIC”. The concept presented illustrates the process (Convert) generated expansion in the form of additional networks. Additional networks are inserted at following times:

- At the beginning of a segment to initialize the step registers on the first scan.
- Before the networks of each step, to determine if it should be executed.
- After the code for all the steps, to reset the transition flags.
- Before each transition, to determine if the transition should be solved.
- After each transition, to determine if the predecessor’s step flags should be
  reset.
- After the code for all the networks, to set the appropriate step flags.

The initial state of the system is given by the initial step, and every SFC must contain exactly one. A transition indicates the condition whereby control passes from a step preceding the transition to a successor step along the corresponding directed link. Each transition has an associated transition condition which is the result of the evaluation of a single logic network referencing the transition coil. The sequence control is given by the following rules:

- The initial situation of a sequence is characterized by the initial step which is in an active state at the initiation of the corresponding SFC.
- The sequence of control of the active steps takes place along the directed links. It is caused by the clearing of one or more transitions.
- Clearing of the transition occurs when the associated transition condition is true and all preceding steps connected to the corresponding transitions are active.
- The clearing of a transition simultaneously leads to the activation of all the immediately following steps connected to the corresponding transition by directed links and to the reset of all immediately preceding steps.

- The alternation of step-transition and transition-step is always maintained in the SFC language. Two steps are never directly linked; they are always separated by a transition. Two transitions are never directly linked; they are always separated by a step.

- If an active step is followed by several transitions (selective branches), the leftmost of these transitions is cleared first, even if two transitions become true at the same time.

- When the clearing of a transition leads to the activation of several steps at the same time (parallel branch), the sequences to which these steps belong are called simultaneous sequences. After the simultaneous activation, the execution of each of the sequences becomes independent.

- One SFC program is allowed per segment.

- You should allocate 1 4XXXX register per step row of SFC program.

9.1.2 Solve order

- If a step is active or any succeeding transition of that step is true, the ladder logic in the step is solved.

- The flag for all transitions is set to false.

- If all preceding steps of a transition are active, the transition is solved. If it is true (the TC element receives power), all preceding steps are made inactive.

- The succeeding steps of all true transitions are set to active.

- The ladder logic in each permanent step is solved.

To implement software functionality that handles the above case definitions, the logic process results in:

- For each scan of the Controller, only those steps that were active at the beginning of the scan are solved.

- A step is executed one more time after it goes from an active to a non-active state. If logic is to be executed only under these conditions, you can program the RSTF block (Read Step Flag), using it like a SENS block. This block will be false only on the last execution.

- Because of parallel links, more than one step can execute in a scan. However, transitions closing parallel paths are not examined until ALL preceding steps are active.

- On each scan all the networks in the permanent step are executed after the normal steps and transitions. Thus, if an action needs to be taken from many different steps, those steps can set a flag, and the action can be performed based on the sum of these flags in a permanent step.

9.1.3 SFC Edit Programs Functions

When entering the Editor for SFC from Offline the Menu Bar reads:

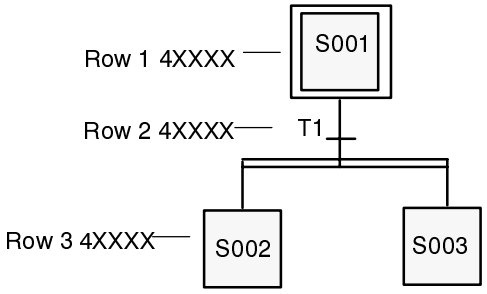
Utility Objects Edit Editors ChgSeg NetList Quit

If Online/Combined the PlcOps function is available

Each segment of a 984 program corresponds to a separate SFC program. The sequence of execution of the segments is controlled by the Segment Scheduler. The current segment number will always be displayed within the window frame. You can change to a different segment. The SFC area is partitioned in a grid of 25 lines and 13 columns. The cursor in SFC is shown as an inverted grid element (the grid cursor). The first row of the chart contains zero or more Permanent Steps, which unlike other steps or transitions are active all the time. In planning your network, keep in mind the Permanent Steps are solved every scan. Permanent steps are executed at the end of the segment.

For ease of maintaining the strict step->transition step->alternation, every line with an even line number keeps only steps and every line with an odd line number (except the first one), keeps only transitions. Steps and transitions are related to certain bits of special 4XXXX registers.

These references are used to keep the actual state (active or inactive) of an SFC element on the controller, so they should be used only for SFC. The networks using these registers are generated when converting the SFC into Flat Structure. The range for these registers is specified in the Configuration under “4XXXX for SFC”. There should be one 4XXXX register allocated per object row of SFC program, as illustrated:



SFC 4XXXX Assignment

9.1.4 Objects

This shows the objects accessible in SFC.

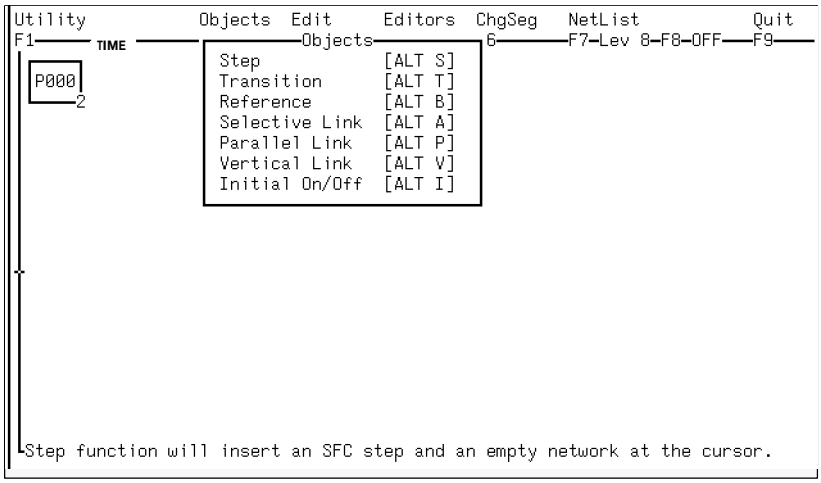


Figure 252 Program Mode with SFC Object Pulldown

9.1.4.1 Step

You can insert SFC Objects by moving the cursor to the desired position and then selecting from the Object Pulldown Menu a step/permanent step transition or reference. All steps get a name beginning with an “S” followed by a number between 0 and 999. A permanent step is inserted in the first line by the step command, it is marked by a “P” followed by a number between 0 and 999

A step is an organizational unit within the Sequential Function Chart. It contains a list of one or more networks associated with this step. The attached networks are executed, if the step is

active. A step is represented by a box containing a step symbol starting with an “S” followed by three digits. A permanent step is represented by a box combining the symbol “P” followed by three digits. Permanent step boxes have no connections. The number of networks in the associated network list is shown in the bottom line of the step box. When inserting a new step this number is one, because one empty network is automatically inserted. There can be no more than 998 networks in a step. If there is exactly one permanent step and no other element in the SFC, this permanent step can contain up to 32,000 networks. If the SFC contains such a permanent step, no other SFC element can be inserted. The shorthand key is <**Alt S**>

9.1.4.2 Transition

The transitions get the name “T” followed by a number between 0 and 999. The steps and transitions must be unique in the chart. If you try to insert the same SFC element twice or if the actual grid is not empty, an error message is issued. A transition controls the flow within the Sequential Function Chart. There is exactly one network related to the transition condition which determines the sequence control (i.e., the changing of the active attribute of linked steps). A transition is shown by a vertical link crossed with a horizontal bar. Transitions are named by a symbol of four characters. The first character is a “T” followed by three digits. The shorthand key is <**Alt T**>.

9.1.4.3 Reference

A reference is a shortcut or jump in a circuit. It represents an implicit link to a step. Only steps can be pointed to/jumped to. The representation of a reference is shown as a vertical line bottomed by a “V”. Below this, that step symbol is displayed, which is referenced. References cannot be inserted in the first two lines. The Step pointed to has an arrowhead located on the left side of the referenced Step Object (box). The shorthand key is <**Alt B**>.

9.1.4.4 Link

This function lets you to draw links from one element to another. The SFC elements are connected by links which specify the sequence control between the SFC elements. The direction of these links is implicitly given (downward or left-to-right). Three types of links are available:

Selective- If one step has several successor transitions or if several transitions have one common successor step, this selective link is marked by a single line horizontal link. . The shorthand key is <**Alt A**>.

Parallel- If one transition has several successor steps or if several steps have one common successor transition, this simultaneous parallel link is displayed as a double line horizontal link. The shorthand key is <**Alt P**>.

Vertical- The Vertical link always leads from the bottom of one SFC element to the top of the one below it; explicit links (i.e., no references) always specify the direction in control flow from

the upper-row elements to the lower-row elements. The shorthand key is <**Alt V**>.

9.1.4.5 Initial ON/OFF

By using this function the initial attribute can be set or reset for a given step. The initial attribute is shown as a double line step box and indicates the start of an SFC program. Only one step per SFC chart can have the initial attribute. To modify, put the cursor on an occupied location in the SFC window.The modification is terminated either by the <**Enter**> key, which makes the changes permanent, or by the <**Esc**> key, which restores the previously entered text. The shorthand key is
<**Alt I**>.

Modifications of textual information are done by the <**Ins**> key, which sets the character cursor to the beginning of the text field.

This screen was produced using the programming objects to implement the “Time” example for SFC.

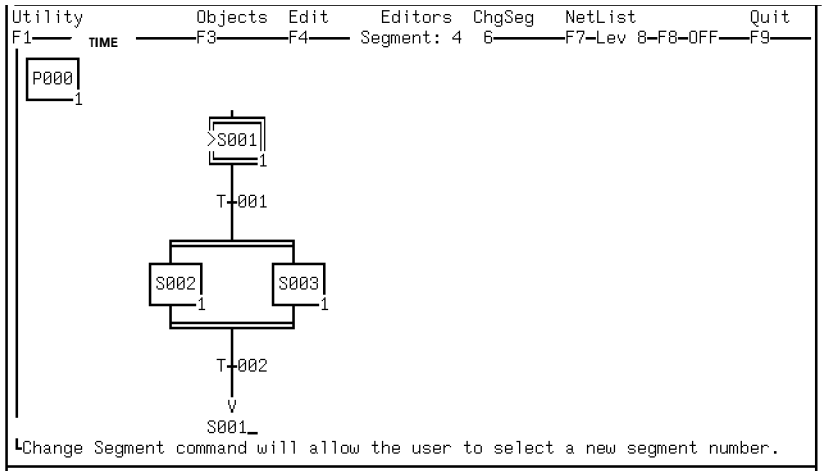


Figure 253 Time Example SFC Diagram

9.1.4.6 SFC Edit Menu

Your “Edit” entry on the SFC Editor Menu line provide you with the following four functions:

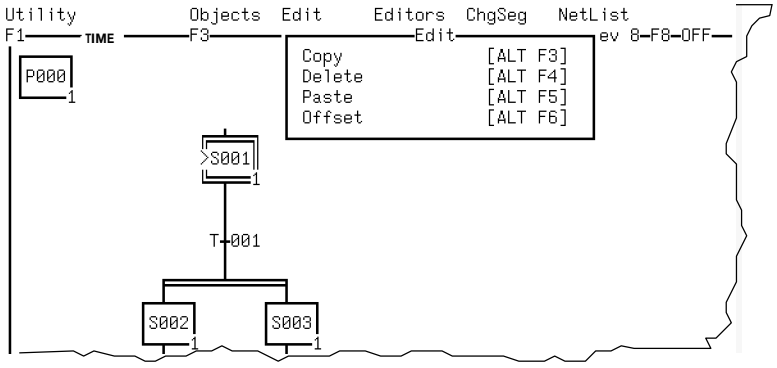


Figure 254 SFC Commands Menu Selection

- Copy

The "Copy" Edit command copies a node, nodes, complete networks, multiple networks or, at most, a complete segment. A copy frame is created by using arrow keys or the cursor keys, < **PgUp** >, < **PgDn** >, < **Home** > or < **End** > keys. A carriage return will copy the highlighted area into a paste buffer. The shorthand command for Copy is < **Alt F3** >.

- Delete

The "Delete" Edit command creates a delete frame by using the arrow keys or cursor keys < **PgUp** >, < **PgDn** >, < **Home** > or < **End** > keys, as above. The frame is displayed in the network window by a highlighted region. When the Enter key is pressed, all networks within the highlighted area are deleted. The deleted networks are placed in a temporary paste buffer in case you wish to recover them. The shorthand command for Delete is < **Alt F4** >.

- Paste

The "Paste" Edit function is allowed only if the paste buffer contains data from a prior copy or Delete command. The contents of the buffer are inserted in the network list at the actual cursor position, and all other networks are shifted to the end with re-numbering. The shorthand command for paste is < **Alt F5** >.

☞ **Note** The Copy, Delete and Paste functions ALL use the SAME buffer.

- Offset

The “Offset” Edit command lets you transform some operands within the network list to other addresses. When in SFC you select a logic step and see to the Ladder for the step then you use the Offset from that menu “Command”. You are prompted for a range of references and an offset value of the modification. If the offset addresses are valid, mark the number of networks to be changed by using the same keys mentioned in the copy and delete commands. The number of replacements is reported when the command is executed. The shorthand command for Offset is < **Alt F6** >.

9.1.4.7 SFC Editors Menu

When you elect an SFC segment for display you have the following Edit functions available:

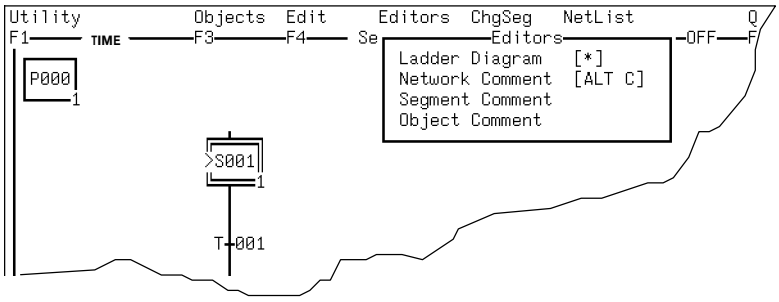


Figure 255 SFC Editors Functions

- Ladder Diagram

  Put the screen cursor on any SFC element you want to see the Ladder representation of, then select Ladder Diagram. The <Esc> key returns you to the SFC previously displayed. The Quick Key is < * >.

- Network Comment

  Put the screen cursor on any SFC element you want to see the Network comment for. The segment and step information are displayed and the editor cursor is placed in the first character position of the highlighted comment field.

- Segment Comment
  Put the screen cursor on any SFC element you want to see the Segment comment for. The Segment comment is displayed with the block cursor highlighting the SFC SEGMENT header. When you press the return key, the character cursor is placed under the first character of the comment and you begin editing if required.

- Object Comment
  This editor is similar to the network Comment Editor and is used for comments of the SFC element (step, transition or permanent step).

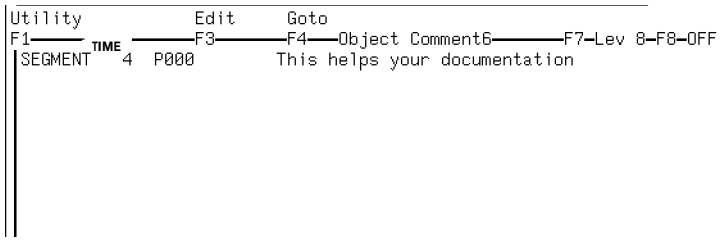


Figure 256 Example of Object Comment Editor

9.1.5 (ChgSeg) Change Segment

This function is available in SFC only and allows you to switch between segments. The dialog window asks you to select the segment number desired, The current segment is changed for the selected segment. The segment number may range from 1 to 32. <**Alt G**> is the Quick Key.

9.1.6 (NetList) Network List

This function operates the same as described in the Ladder Edit description in the prior chapter.

9.1.6.1 SFC Quick Key Summary

The following shorthand is provided by the SFC Editor:

<**Alt S**>: insertion of a step/permanent step at the cursor position

<**Alt T**>: insertion of a transition at the cursor position

<**Alt B**>: insertion of a reference at the cursor position.

<**Alt I**>: Set Initial On/Off

During the creation of links, you can use the following abbreviations:

<**Alt A>** or the shorthand key **-** : for a single line horizontal link

<**Alt P>** or the shorthand key **=** : for a double line horizontal link
<**Alt V**> or the shorthand key **|(bar)**: for a vertical link

The link function is implemented as a toggle, so the same function is used to insert and delete a connection. Horizontal links cannot be inserted in the first line and in the last column. Vertical links cannot be inserted in the first two lines.

9.1.7 Permanent Step Online Programming

Additional code is generated for controlling the SFC structure. The next figure shows the Online mode view of the step 1 transition (TC) as it is expanded to function in the controller. The fact that the network is skipped is printed in the upper right corner of the screen. Also shown is another of the converted networks, this one comparable to the original source and shows a brief heading indicating this is SEG 3 #(Net) 8. ***When in the Online Mode,the SFC structure is reducedto one Permanent Step for each segment which allows the networks to be viewedin the Flat Ladder Logic format..*** The assigned networks cover the complete program for the given segment.

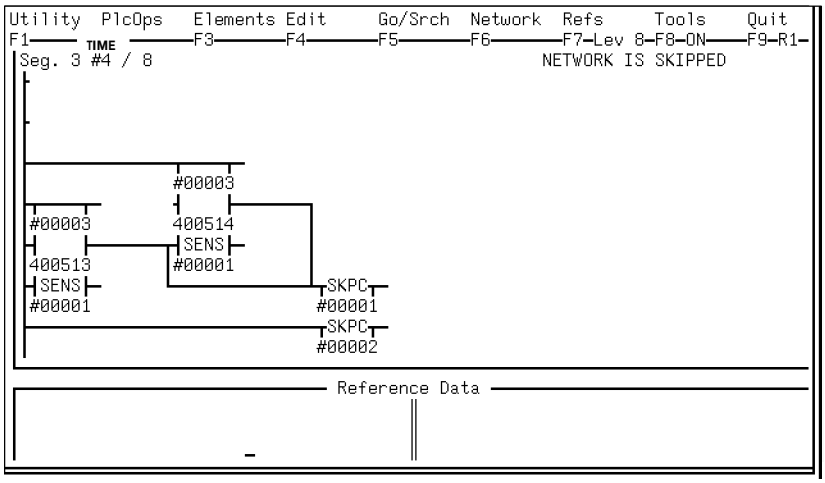


Figure 257 Example of Online Mode "Flat Ladder"

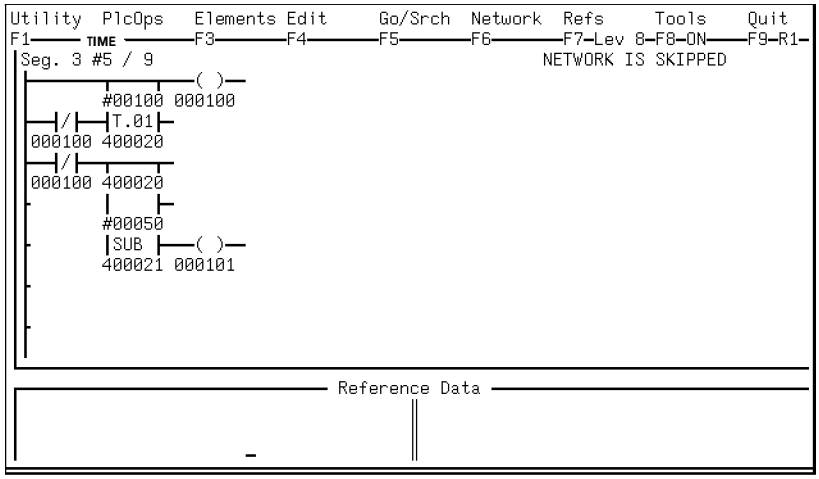


Figure 258 Part of the PLC Converted network Showing SFC Ladder Logic

9.1.8 Combined Mode Programming within SFC Structure

For ease of debugging, you can only access those networks you have programmed as the access is handled via the single network lists of the SFC elements. Thus, the SFC control code cannot be accessed in the controller. Insertion and deletion of actions is possible only when working in a Permanent Step. Otherwise, the operands of a SKP function would not fit the actual situation as the generated SFC code is based on this operator. It is up to you to ensure that the program's semantics are not changed. The next section describes the Combined Mode's Programming functionality of SFC Editor. Network list Editor and Ladder Diagram Editor appear later in the chapter.

The <**PrtSc** > function of IBM-ATs and compatibles is not disabled. Thus, a screen can be printed easily for debug or documentation.

9.1.9 SFC Editor

The SFC Menu When in Combined Mode has no Object Or Object Edit capability but unlike Offline there is a PlcOps entry. There are three functions under PlcOps:

1) Start
2) Stop
3) Single Sweep

Changes within existing Steps/Transitions are allowed by selecting the appropriate Ladder network, but increasing or decreasing the number of networks is not allowed. Only within Per-

manent Steps can networks be added or deleted in Combined mode. Active steps and transitions are highlighted with the controller in RUN mode. This is because the program on the controller has been generated by converting the SFC program. If this SFC program is changed in Combined Mode, the SFC program has to be converted and transferred from panel to PLC again. Therefore, in addition to the Utility and PlcOps commands, only the following SFC Menu commands are available:

Editors ChgSeg NetList

Chapter10
ASCII Programming

- Create ASCII Messages
- Formatting Techniques
- Editing Features

10.1 ASCII Overview

The Main Menu Offline/Online and Combination selections offer editing features for ASCII messages.

ASCII tends to be a stand-alone messaging technique allowing messages and ASCII commands to be routed to ASCII devices.

Messages are created, edited and displayed in a temporary buffer. Leaving the ASCII editor preserves the messages being worked on in a temporary file. Before a Modsoft session is complete ***you must Save*** the edited file to store the messages on the hard media permanent file.

10.2 ASCII Editor

The ASCII Editor is used to create, edit and simulate ASCII messages. The ASCII message file you create on the panel is Transferred as ASCII messages **to** the controller. On the other hand, an ASCII message file can be created through a Transfer operation **from** the controller. The ASCII Menu contains the following items:

Utility PlcOps Objects Edit Msg I/O Quit

The PlcOps function is **not** available in Offline.

The ASCII Editor is not a free text editor, so only syntactically correct ASCII messages can be edited. This is done by using specific Objects Commands. In general you edit a message by moving the cursor to the mentioned position and calling the corresponding object.

Messages of any combination of objects cannot exceed 256 characters, while fields within messages are limited to 128 characters. Each field has a three word overhead associated with it which is displayed as the initialized count of the display field “Length:3” . The Modsoft display will also indicate the “Number of words Used: ” as a deduction from the base allocation in the configuration section, and indicate for you the “Number of Words Free: ” remaining. Modsoft will insert a comma as a field separator.

10.2.1 Objects

An ASCII message is a list of fields, where each field is delimited by the single quote (**'**), terminated by a comma, and does not exceed 256 characters (bytes). When constructing a message, the running calculation will include the number of characters (in words, 2 characters per word), 3 overhead words and one word for the delimiters.

The special format of each field is automatically created by the object function. According to the selected mode (Insert/Overwrite), fields are inserted or are overwritten at the current cursor position. There is a Delete function to delete the field at the current cursor position. The deleted field is filled to a paste-buffer and may be pasted later. The following objects are available.

10.2.1.1 Text

An arbitrary ASCII string, enclosed in single quotes (generated by Modsoft, e.g., ‘message string’). The Quick Key is <**Alt T**>. The following illustrates ASCII Message text entry. This data will be “Listed” as part of the Time example and has already been provided in the configuration.

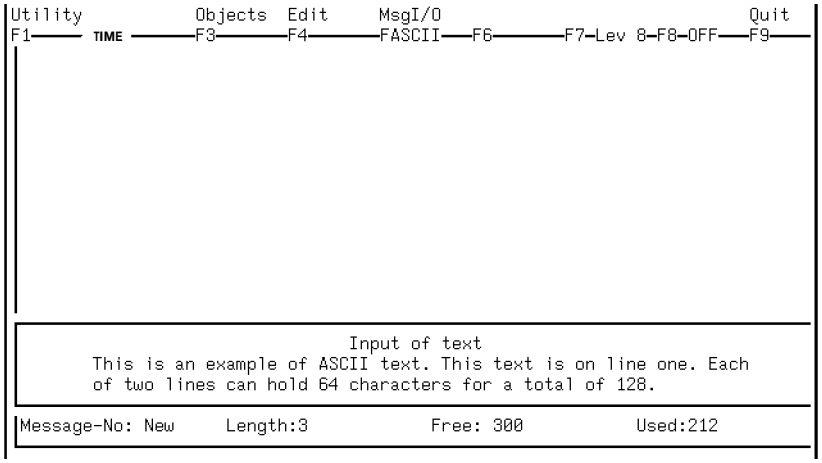


Figure 259 ASCII Text Input Area

The process required to create the message is:

Step 1 Go to the “Objects” menu bar selection and select “TEXT” from the sub menu

Note that you type in the text and Modsoft puts in the special delimiters.

After terminating the text as illustrated it will appear on the top line of the display with delimiters in place, and the number of configured words used is calculated and displayed on the bottom of the screen.

Step 2 If finished with the message, return to the ASCII menu bar, select “Msg I/O” and assign a message number via the “Save Message” function.

Step 3 To continue; go to “command”, clear the screen and resume text entry.

As an example of input and memory utilization consider the following:

Type in the text field QUERTY
When you enter the message it will appear at the top of the message screen with the terminators put in for you, i.e., ‘QUERTY’
The length count is equal to 7.

10.2.1.2 Control Code

A 3-digit ***octal*** control character (in the range 000-377), for which Modsoft supplies the double quote delimiters, e.g., “033”, “014”. The Quick Key is <**Alt C**>.

10.2.1.3 Variable

A variable field of the format Number of registers, Type and Field Length, where the “Type” letter selects:

A	Character	B	Binary
H	Hexadecimal	I	Integer
O	Octal	L	Integer with leading zero’s.

“Length” is an integer repetition factor for the field, and “number” stands for the number of registers, e.g., 2H3 as an input stands for 2 registers, each containing 3 Hex numbers. The simulated display of this format would be : HHH HHH . The Quick Key is <**Alt V**>.

10.2.1.4 Space

The ASCII message symbol for space is X. A series of spaces can be specified by a number between 1 and 99. The Quick Key is <**Alt P**>. For example, if you have two fields that need separation, entering 1H2,5X,1H2 you will create the format for two registers each with two hex character pairs with 5 spaces between them. A simulated display looks like: HH HH.

10.2.1.5 CarriageReturn

A ‘Carriage Return’, encoded by the “***/***” symbol is inserted. The Quick Key is “***/***”.

10.2.1.6 Flush

This Edit selection provides 4 different command format structures that condition the P892 input port buffer data. You can specify how to clear a message buffer or screen by sending command codes. You are prompted for one of the four function types as seen in the next figure. The delimiters **< >** are inserted by Modsoft and the programmer can enter any of the following (the Quick Key is <**Alt F**>):

□ **0**

This format requires 1 register and is intended to ***remove all*** characters currently in the buffer, which has a data field width of 256 characters (bytes) and can contain data values in the range 0 to FF Hex. For example:
Buffer contents 1 2 3 5 6 7 9 0 before
- - - - - - - - - after

□ 1 ; bbb

This format requires 1 register and is intended to remove (b) the number of characters specified, from the P892 port input buffer. The value of (b) can be from 1 to 255. For example:
Buffer contents 1 2 3 4 5 6 7 9 0 before (where b = 5)
- - - - - 6 7 9 0 after

- 2 ; hhhh

This format is intended to flush the buffer ***until a match*** is found for the flush terminator character pair, then stop, NOT flushing the match characters. This is a 2 register format. The first contains just the type identifier (2), while the second contains the hexadecimal values of the terminator pair, in the range 0000 to FFFF, for which the the buffer will be searched. If the second character of the match pair is not a NUL (00), then the next character in the buffer must test equal or the search is resumed. If null or equal to the final character, the flush is performed up to but not including the matched terminators. For example, with the search pair TE:
Buffer content 123ATEST567ATEST before (where begin/end pair are 5445 hex)
----TEST567ATEST after

- 3; rrr; hhhh

This format allows you to flush specific groups of data as supplied by the value of “r” which can be from 1 to 255. The begin/end character match pair function as described above. The match and flush will be performed as many times as specified by the value “r”. This command form includes flushing the match characters. The format takes 2 registers, one for the type and number of times to execute and the second for the hex value of the match pair. For example:
Buffer contents 123ATEST567ATEST before (r = 2 and begin/end = 5445 hex)
------ST------ST after

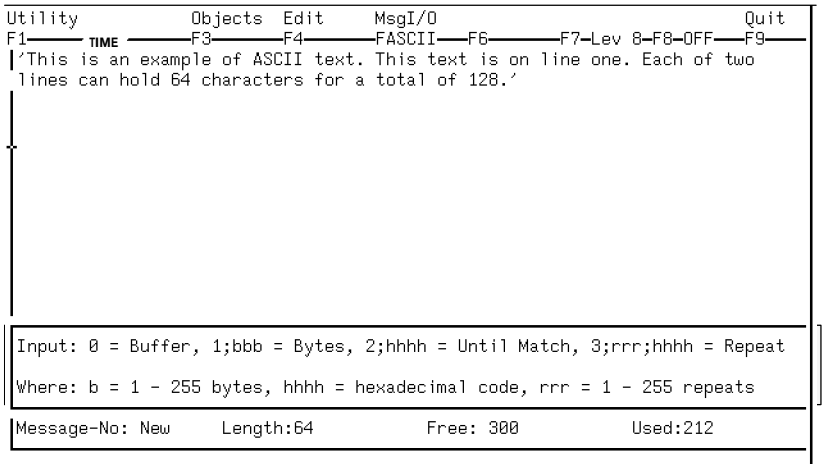


Figure 260 ASCII Flush Buffer CommandWindow

10.2.1.7 Repeat

You can specify the repetition of several fields. Nesting of repeat brackets is not possible. The Quick Key is <**Alt A**>.

Example:

6 ('Item', I2, 4x, I5, /) will produce 6 lines, each containing the fields 'Item', I2, 4x, I5 and a <**CR,LF** >.

10.2.2 Edit

10.2.2.1 Copy

The contents of the field at the cursor position are copied to the paste buffer. The field itself stays unchanged. The Quick Key for copy is <**ALT F3**>

10.2.2.2 Delete

With this function, ***the field*** at the actual cursor position is deleted. The contents of this field are stored in a paste buffer. The Quick Key for delete item is <**Alt F4**>. While this function deletes the field displayed on the screen, it does not delete the field held in memory. If you make a mistake the entire message can be reloaded via "Load Message" as long as that message was previously saved.

10.2.2.3 Paste

Depending on the mode, the deleted/copied field is in the paste buffer. The insert is performed at the cursor position, and overwrites the field the cursor is positioned on. The Quick Key for paste is <**ALT F5**>.

10.2.2.4 Clear Screen

The message currently on the screen is deleted from the screen. A warning is displayed if there are any unsaved entries.

10.2.2.5 Simulate

The currently loaded message is displayed as it would appear as if being driven to a standard ASCII terminal. This is especially useful for debugging ASCII message formats. For example, to develop a format for the following number set:

123 456
789 147

Make the format entries as in the top screen and review the work with the Simulate function as in the bottom screen.

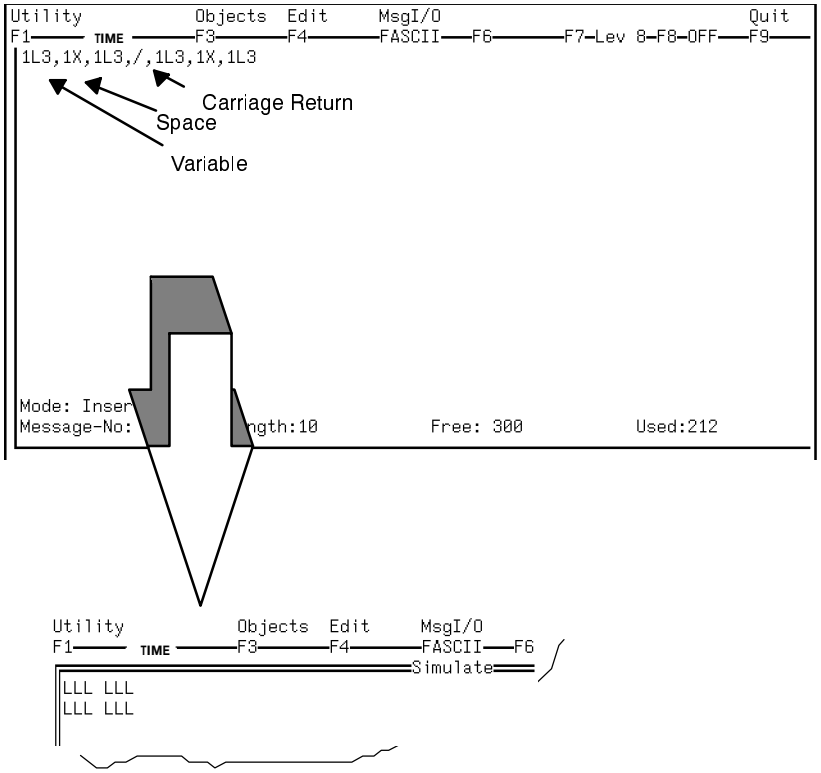


Figure 261 Format Developmentand View via Simulate Screen

During the development, you can make use of the insert mode by placing the cursor on the construct line item, which the insert should precede, and then selecting the Object of choice.

10.2.2.6 Used Messages

The next figure Shows the number of the messages which have already been saved or uploaded. This figure also slows the actual message content.

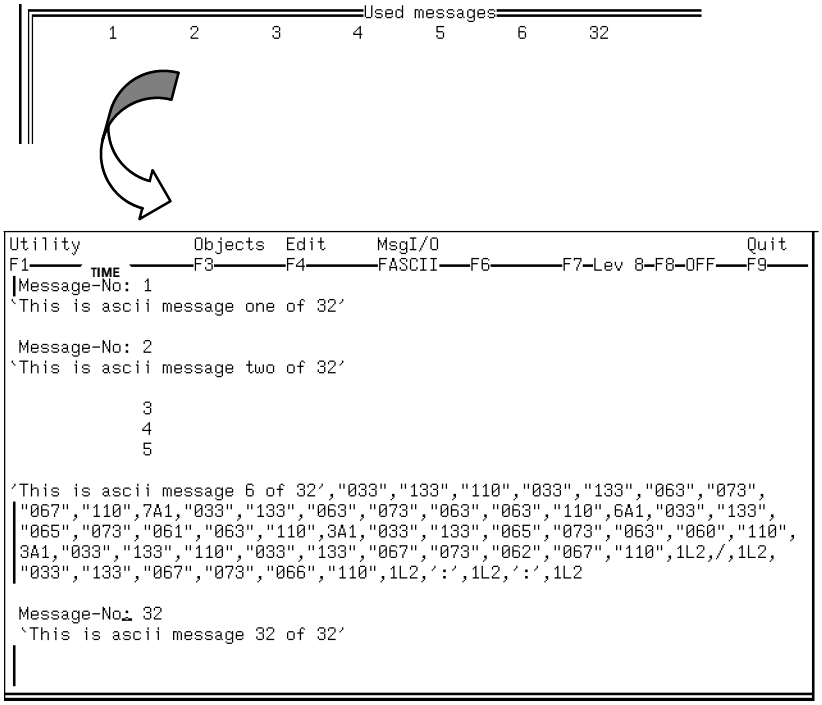


Figure 262 Used Message Screen

10.2.3 Msg I/O

To edit an ASCII message, you must create a new message or load an existing one. After the edit session, the ASCII messages can be saved. Before leaving this menu, there is a warning if you made changes without saving them.

10.2.3.1 Load Message

Load Message prompts you for the message number, you want to edit. If the message exists, it is displayed on the screen. If not, an error message is displayed. The shorthand is <**Alt L**>.

10.2.3.2 Save Message

You are prompted for the message number and the currently edited message is written to the ASCII Message File. If the number is out of range, or the message number already exists, a warning is displayed. The shorthand command for Save Message is <**Alt S**>.

10.2.3.3 Import

You can specify a file that contains the ASCII message text for all messages in standard text format.

10.2.3.4 Export

You may specify a file to transfer panel or PLC messages into a standard text format file. All messages are stored to a file that can be accessed by a standard ASCII Editor. To demonstrate import/export the following example is provided.

The messages 1-6 and 32 exist in your "Time" example.

If you create a message 7 that reads - export example - save it so that a message # ID becomes associated with it, then select export and give the full path C:\Modsoft\Programs\export.

The file export is created when you save the ASCII edit via the ASCII Editor "Msg I/O Export" function. The exported format looks like that illustrated, where the message number occupies one line starting in character position 2 and the message delimited by single **'** and terminated with a comma **,** starts at line character position 1 of a subsequent line.

```
_ 1
'This is ascii message one of 32',
  7
'export example',
  32
'This is ascii message 32 of 32',
```

Figure 263 Export File and Format

If you were to change the file 1 "Export" and execute the import function with full path name followed by load message, message 7 is now the altered message as shown below.

```
Utility         Objects  Edit     MsgI/O                                Quit
F1------ TIME ------F3-------F4-------FASCII----F6-------F7-Lev 8-F8-OFF----F9----
 'Export Example modified by Dos Edlin'
```

Figure 264 Import of Altered Export File

10.2.3.5 Delete One Message

A message specified by a message number, can be deleted in the message file.

10.2.3.6 Delete All Messages

All saved messages will be deleted from the message file.

10.3 ASCII Online Differences

All functions of the ASCII Editor are available in ONLINE COMBINED or OFFLINE Mode. Only the “Load”, “Save” and “Delete” commands work differently.

- Load

LOAD prompts you the message number you want to edit. If the message exists in the controller, it is loaded from the controller and displayed on the screen. If not, an error message is displayed.

- Save

The currently edited message is written to the controller as well as to the database. You are prompted for the message number. This number must not exceed the number of configured messages. Before saving a message, a check is done to determine if there is still a message with the specified message number. If so, you have to confirm the replacement of that message. If the number you entered is out of range, an error message is displayed. This command can be executed if the controller is in Run Mode or in Stop Mode. The shorthand keys are <**Alt S**>.

- Delete

If in Combined mode, a message specified by a message number is deleted on the controller as well as in the database. When in this mode, the message number is deleted in the controller. If the specified message does not exist on the controller, an error message is displayed. This command can be executed if the controller is in Run Mode or in Stop Mode and is available Offline. Additionally, a new function is available:

Used Messages - When this function is called a window is opened showing all messages actually loaded. Strike any key to remove this window from screen.

☞ **Note:** The Message area size is not adjustable Online

10.3.1 PlcOps

Of the PlcOps commands previously defined, the following functions are allowed in Online of Combined programming modes: Start, Stop, Single / Constant Sweep and Set Hardware Clock.

Chapter 11
Macro Programming

□ This chapter instructs you how to use the Modsoft Macro Editor.

Note The SFC flag must be turned to **ON** for Macro Programmed elements to be used in a Ladder Logic program.

11.1 MACRO Programming

The Macro Editor is selected from the Main Menu Tools pulldown. The Macro editor provides you with a method to define a sequence of common code that can be called to execute in various networks without the requirement of re-writing the code each time used. The Macro consists of two parts: The Macro Header and the Macro Body Block.

The Macro Header contains the Macro name, comment and all the reference parameters (preceded by a $ sign automatically inserted by Modsoft). These are used within the Macro body, plus associated comments for each parameter. When a Macro is converted to flat ladder logic, all parameters are type checked and the ladder logic is replicated for each use of the macro.

The next figure illustrates the created macro used in the “Time” example macro. In the upper frame, the Macro editor is used to create the Macro header Parameters. This frame shows the name assigned to the macro and also illustrates the Macro comment field above the Parameter entry area i.e. “Timer” The lower frame shows the ladder implementation of the Macro call (using the commands Macro selection) that equates the **$** parameters with register values.

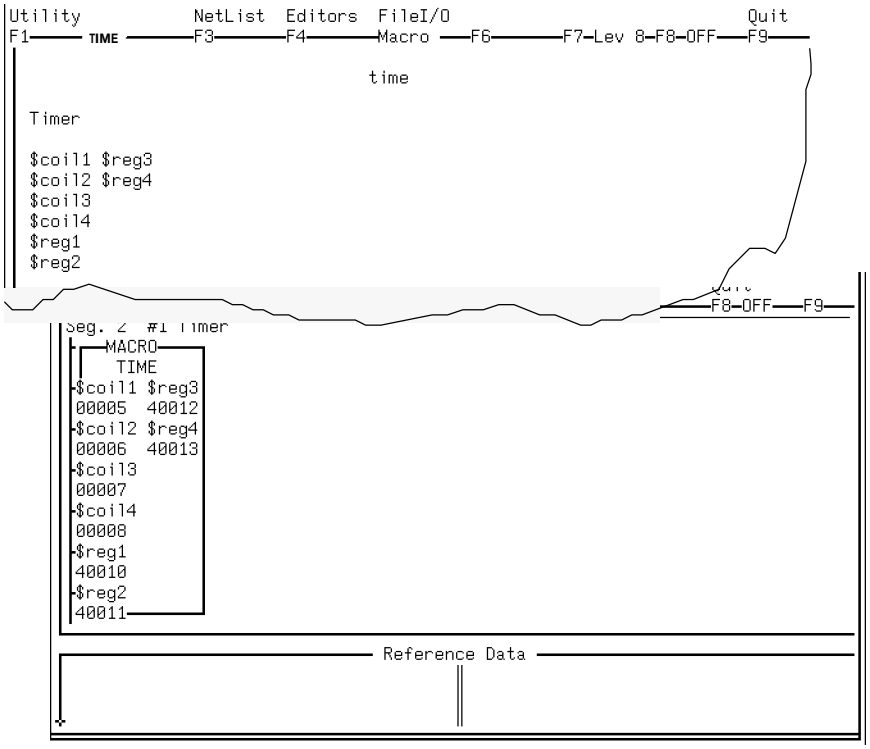


Figure 265 Example of Macro Header and Logic Call Block Used in ”Time”

The Macro Body Block consists of Symbolic network references. You can insert, delete, copy and paste networks and create the corresponding ladder diagram via the Editor to program the macro networks. Networks can be fully Commented by selection from the Editor pulldown menu.

☞ **Note** Recursive macro calls are not allowed.

When the Macro Editor is entered, either a macro header name or *NEW* is displayed. The block body for a given macro is reached via the Ladder Command Macro entry. The cursor keys are used to highlight the formal parameters and to position after the last parameter.

The Macro menu bar contains the following:

Utility NetList Editors File I/O Quit

11.1.1 Parameter Description

The Macro header contains a maximum of 66 parameters: 6 lines of 11 parameters each. These parameters are displayed consecutively from top to bottom as entered; there are no empty spaces. If there is existing data, a list of parameters is displayed when selecting Macro from the “Editor” menu entry.

☞ **Note** For parameters to macros, use registers, not constants during your development. By using registers, you can change values while online to the controller. Macro parameters alone cannot be changed while online. Once you have determined the precise value to use in one Macro application you can edit the call of the Macro in offline mode and use a constant.

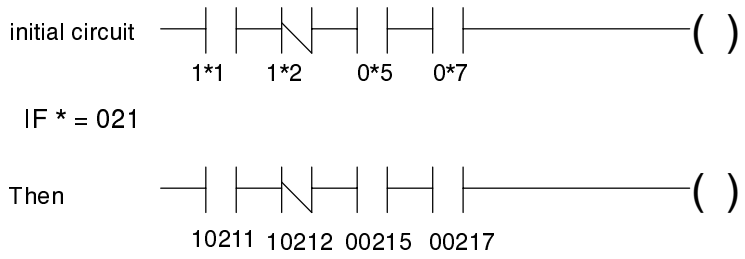
11.1.1.1 Insert

Use the cursor movement keys to put the cursor below the location of the desired insert. When you strike any alphanumeric key, the macro symbol “**$**” and the first of up to 5 characters are displayed. The parameter name chosen must be unique. Only one parameter can be assigned in a Macro name **“ * ”** which is used as a “wild card character”. When defining the parameter *, you will need to create a dummy parameter line, then delete <**Back Space**> over all characters including the $, to allow the field to hold the * symbol. This is due to a conflict

with the Ladder Diagram shorthand key. When you continue to develop the Macro body the same process might be needed depending on what location in the block the * is being placed.

☞ **Note** Where an element has a standard field width, the * indicates more than one character if used that way (i.e., filling a reference field that defaults to **?????** with “0*1” means the * equal to 3 character positions). Length is checked on conversion.

The above character substitute would create the following situation:



Wild CharacterExample

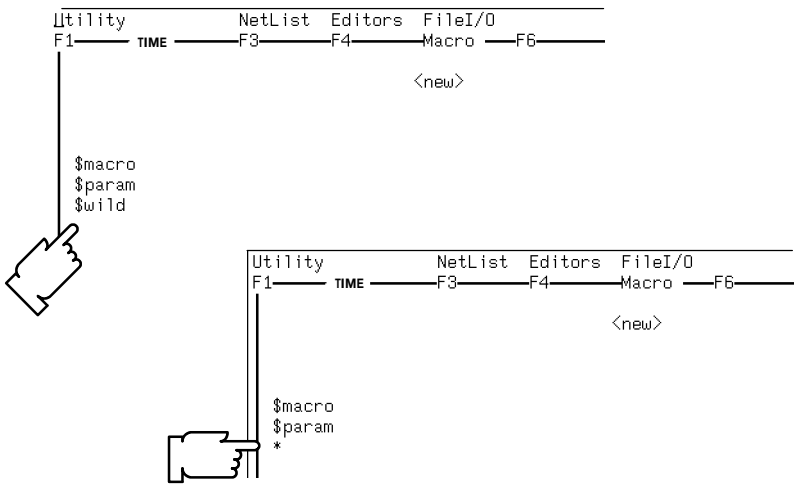


**Figure 266 Implementing the “wildcard Character” Macro Parameter *

11.1.1.2 Delete

You can delete a parameter by moving the cursor to the desired position and selecting the **<Del>** key. All successive parameters are shifted up one position.

The parameter selected is modified by a combination of insert and delete entries and is terminated by pressing the < ↵ > key, or to restore the original without change, by pressing the < **Esc** > key.

11.1.2 NetList

This entry from the menu bar switches you to the network list for the macro. The network list window is opened and the following menu is displayed:

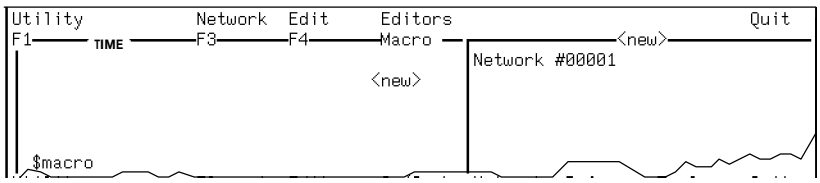


Figure 267 Netlist Menu Display

11.1.3 Network

In an initial setup i.e., no networks exist, the cursor is positioned in the network window and a ***new*** empty network is inserted. This new network has no symbol and no comment assigned. You can use the edit functions discussed in previous Chapters to edit networks by Insert, Copy, Paste, and Delete.

11.1.4 Edit

11.1.4.1 Copy

The “COPY” command copies one network, or group of networks, in the network list. A copy frame is created by using the cursor keys, < **PgUp** >, < **PgDn** >, < **Home** > or < **End** > keys. A carriage return will copy the highlighted area into a paste buffer. The shorthand command for Copy is < **Alt F3** >.

11.1.4.2 Delete

The “DELETE” command creates a delete frame by using cursor keys < **PgUp** >, < **PgDn** >, < **Home** > or < **End** > keys, as above. The frame is displayed in the network window by a highlighted region. When the Enter key is pressed, all networks within the high-

lighted area are deleted. The deleted networks are placed in a temporary paste buffer, in case you wish to recover them. The shorthand command for Delete is < **Alt F4** >.

11.1.4.3 Paste

The “PASTE” function is allowed only if the paste buffer contains data from a prior Copy or Delete command. The contents of the buffer are inserted in the network list at the actual cursor position, and all other networks are shifted to the end and re-numbering. The shorthand command for paste is < **Alt F5** >. You can copy or delete networks from the standard Ladder and Network List editor and Paste them into the Macro editor.

11.1.4.4 Offset

The “OFFSET” command lets you transform some operands within the network list to other addresses. You are prompted for a range of references and an offset value of the modification. If the offset addresses are valid, mark the number of networks to be changed by using the same keys mentioned in the copy and delete commands. The number of replacements is reported when the command is executed. The shorthand command for Offset is < **Alt F6** >.

11.1.5 Editors

The Editors selection from the menu bar produces a two level pulldown menu allowing the selections for Macro Ladder Diagram or Macro Network Comment. The shorthand keys are < ***** > and < **Alt C** > respectively.

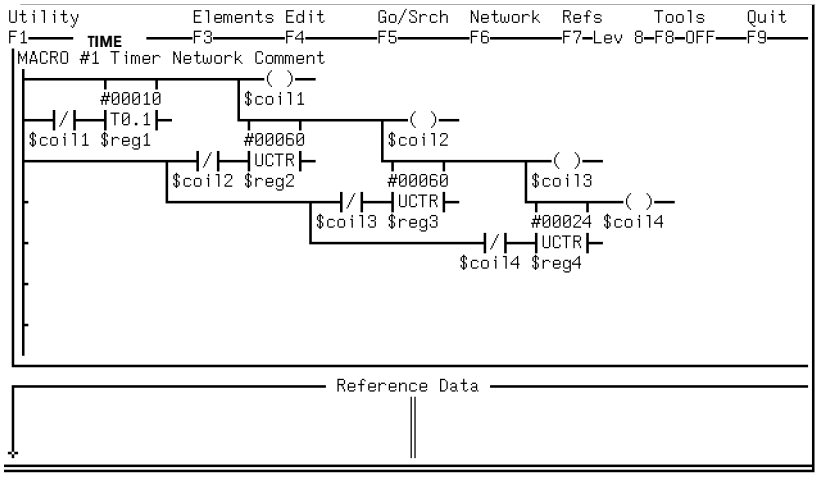


Figure 268 MacroLadder Logic Used in “Time” Example

The Macro Editor created Ladder Diagram is much the same as the regular Ladder Diagram Editor.

You can select any of the Macro Ladder functions in the same way as in the LDE, including most of the shorthand notation. The primary difference is that the Macro ladder editor can assign defined parameters to reference nodes. To do the parameter assignment, place the Ladder cursor over the node and type $param (where “Param” is a predefined macro parameter).

The Macro Network Comment is similar to the Program Network Comment Editor. You can assign a comment to the Macro itself. When selected, a screen appears with a title area of 32 characters and a screen area of 20 lines in which to accommodate the Macro comment.

11.1.6 File I/O

When entering the Macro Editor for the first time, no macro data is present. You begin by creating a new macro, and are so instructed by the word < NEW> in the display. Using the File I/O functions, you can either create a new Macro or load an existing one from the file system. After an edit session, you save the macro to store the changes in a disk file. Load, Save and ClearMacro are available from the File I/O menu pulldown.

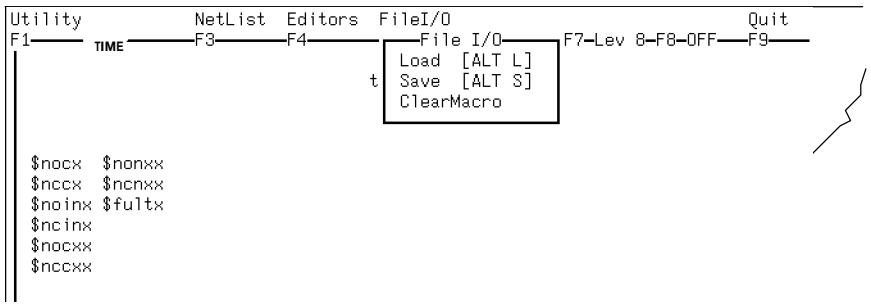


Figure 269 Offline Macro Menu File I/O pulldown

11.1.6.1 Load

When “Load” is selected, either type in the macro name or get the list of macros which are present in the macro directory via the <**?**> key. After selection, the file is loaded into the Macro editor. If you enter a name of a non-existing file, a new macro is assumed and will be created. The shorthand for Load is < **Alt L** >.

11.1.6.2 Save

The Save selection prompts you for the name of the macro. Modsoft defaults to previously loaded/created macro name, which can be accepted by pressing the <↵> or the name can be changed. The shorthand key is < **Alt S** >.

11.1.6.3 Clear Macro

Selecting the "ClearMacro" displays a user prompt to verify your selection with a Yes or No response. A "Y" response results in the macro editor re-initialization to its initial state, with the exception of the paste buffer (which continues to hold its contents).

11.1.7 MACRO Application

The principals presented in the preceding paragraphs can best be summarized by an application example. First, lets state the problem.

- A machine has three major functions controlled by solenoids that are driven by Triacs. The requirement is to utilize contacts to monitor the state of the triacs and signal a failure.

A circuit devised to provide the monitoring required is illustrated for you. Although not a complex circuit, the steps you perform to write a macro (so you only have to enter the circuit once) are the same as required for more complex circuits where the macro saves considerable time.

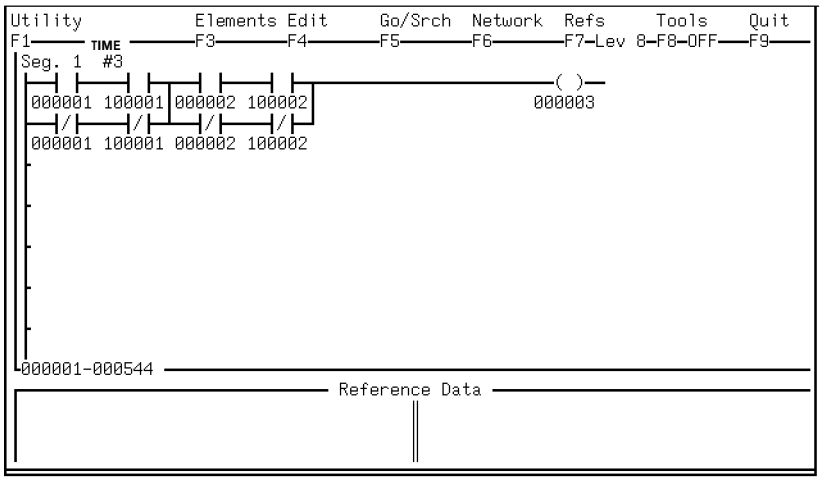


Figure 270 Example Circuit for Triac Monitor

☞ **Note** Before proceeding verify that your configuration (Overview) has SFC turned ***ON .***

11.1.7.1 Macro Implementation Procedure

The solution to the problem begins when you go to the “Offline” menu and select “Macro” entry from the Tools pulldown and proceed as follows:

Step 1 Select the program File set (from the displayed names). This is the program your macro will be associated with.

Step2 The screen has a temporary name <NEW> and a blank screen. As you enter the first Header character it is automatically preceded by a $ symbol. You are allowed 5 characters after the $.

The next figure illustrates the parameters associated with the circuit. In this example, the parameters are simply noted as normally open or normally closed and the coil is labeled.

Step 3 When you press the <Esc> key at the end of the Header entry process, you are prompted as to whether you want to save it Y or not N. If you elect the save, a filename prompt and entry field is displayed with the Macro extension of .MCR

If you have a file by the same name you are given the option to exit the process or overwrite the existing file.

Step 4 Re-select the fileset from the Main,Tools,Macro entry point, and position the menu cursor on the Editors entry.

Step 5 Select Macro Ladder Diagram from the Editor menu pulldown list. The display provides the selections that you use to continue with the Macro development and its commentary.

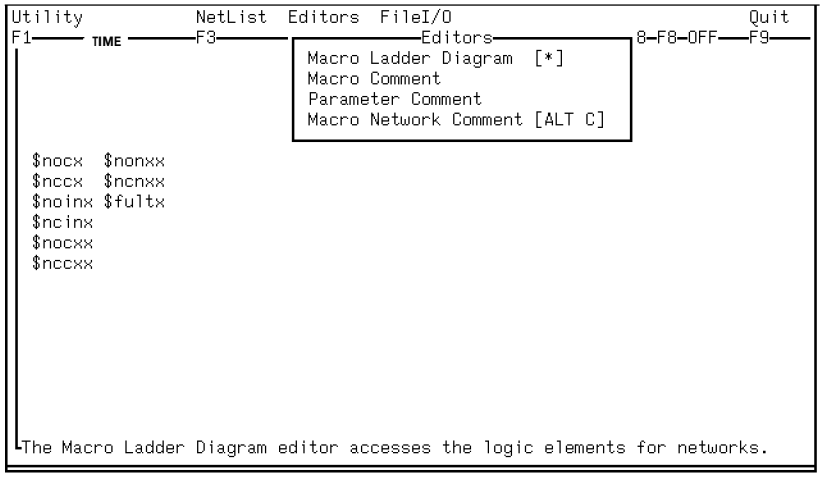


Figure 271 Macro Selection Screen With Editor Sub Menu

Step 6 You now create the Macro specific Ladder Logic with the relationship to the Header $Parameters previously entered. Your diagram looks like:

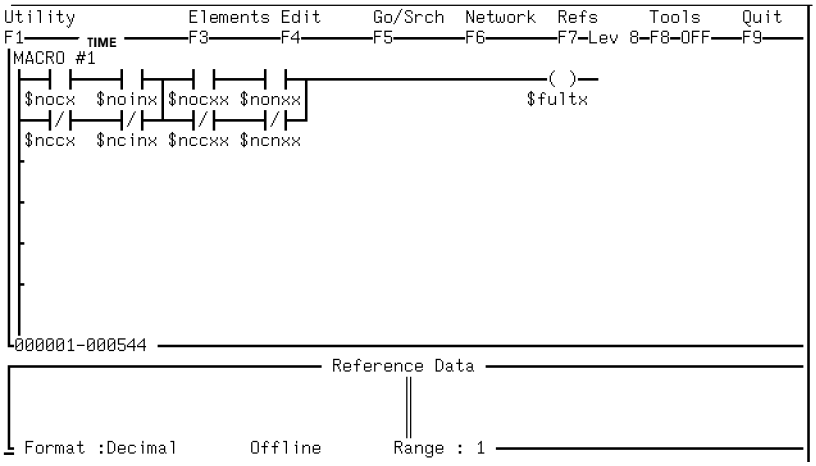


Figure 272 Macro Development Ladder Diagram

Step 7 When the Ladder/Reference is completed, press the <Esc> key which saves the logic just completed and returns to the Macro entry level screen. When you leave this screen you are prompted for a name for the file .MCR before leaving the editor. If the file exists the overwrite? message is displayed and after your response the cursor is returned to the Main, Tools, Macro entry.

Step 8 Implementation of the Macro relationships in a PLC program call is performed under the "Offline" menu " Select Program" selection. When the segment status is displayed set the cursor in the segment you want the macro call block to be developed in and press the return key.

Use the Page down key and Network Insert commands as necessary, and display an empty network. Press the ,<Tab> to get to the menu line, select the Macro element and fill in the reference information. One Macro block is recommended per Network. To proceed you enter the information as illustrated.

The **result** achieved thus far is that you would have the first of the three required circuits when you download to the PLC.

Step 9 You create the other two circuits (Offline) by using the copy and offset programming tool commands. Thus with a minimum of keystrokes the logic is replicated. First ***Copy*** the Macro block then ***Paste*** into the next two networks then create offset references such that there is no overlap, and you are within the configured range.

The **resu**lt now is the three ladder logic circuits are created when the macro calls are invoked at File to PLC conversion time.

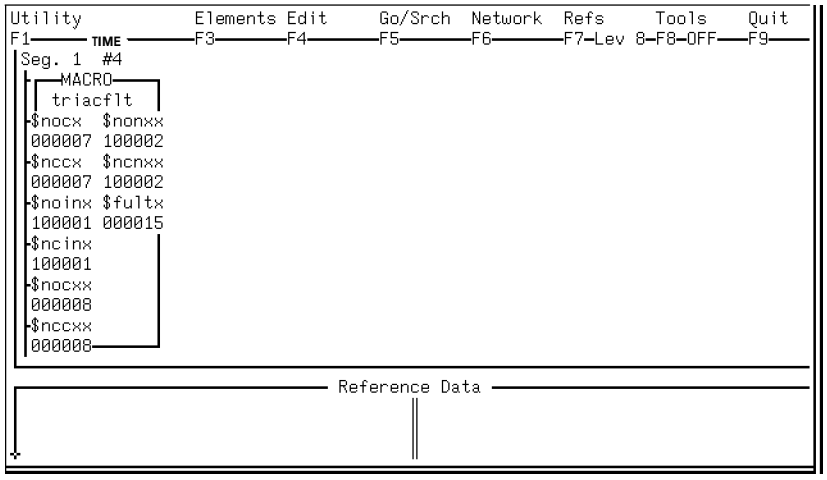


Figure 273 PLC MacroProgram Call Element

11.1.7.2 Online macrorepresentation.

There are two online modes - Online - Combined . If you select Combined you will be working with a representation of the panel program, converted to PLC form and displayed in the form of a macro call block with the state (**E** nable **D** isable) of the references displayed.

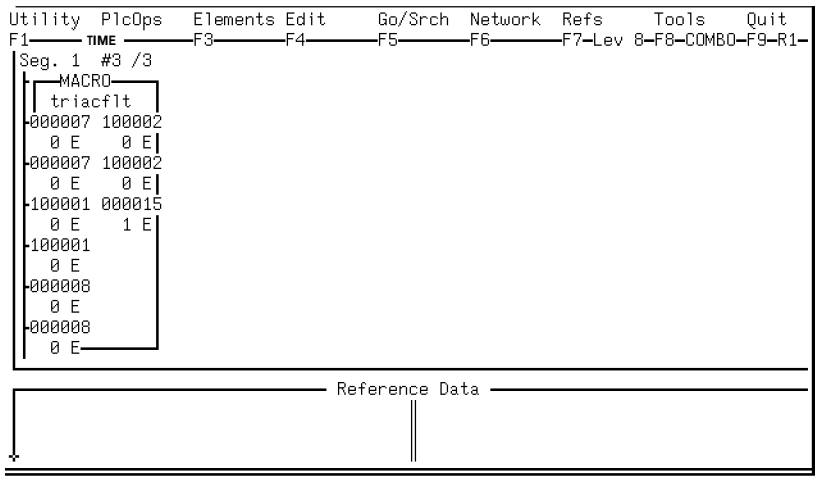


Figure 274 Combined Mode Macro Call Block Typical Display

If you select the Online Direct to PLC mode the PLC based logic is uploaded into the panel and you will see that the macro did indeed create three networks. although on different pages, this figure shows all three where you can see the replicated result of your macro and the offset of the registers.

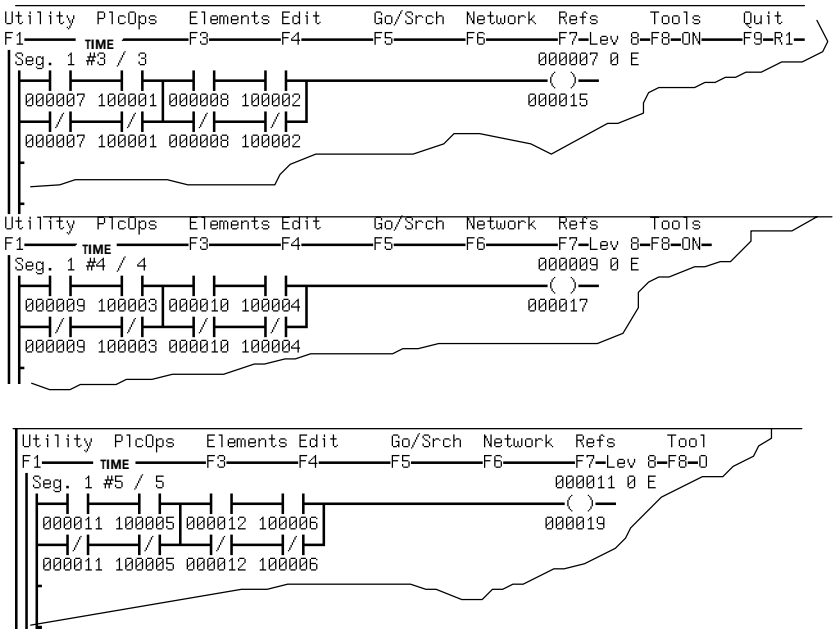


Figure 275 Online Results of the Example Macro

Chapter 12
Print

- This Chapter informs you about the functions and features provided by the Modsoft Print application and associated UPF selections. The application is customized so you can shorten the listing when you are in a development mode and only require specific documentation.

☞ **Note** The new Printers.pdb may contain changes, from the previous release. These changes are not automatically reflected in your modsoft.pdb. There is no affect on your current printing with modsoft.pdb. However, if you edit your printer preferences using the UPF editor, you may encounter compatibility issues between the databases. If this situation occurs, and you need to change the definitions or your device, you have to re-define the printing device that has the conflict, after removing it from the modsoft.pdb database.

12.1 Listing Your Ladder Logic Program

There are ***Two*** elements required to properly list your programs:

1. You must initially customize your printer output to the printer available at your location. Modsoft makes this selection nearly universal by allowing you to set the parameters for all common printing devices and most of the less common Printers. You are allowed to print to a local printer, construct a file for later printing, or print directly to a remote printer.

   This section describes the printer configuration tool that is included in the distribution of Modsoft. Specifically this is the User Preference File Editor (file name UPF.EXE) found in the Modsoft directory.

2. Once the Printer device setup and device selections are completed, you may use them whenever listing programs from Modsoft In this case you will use the Modsoft Main Menu **Tools** Function with the **Print** selected.

☞ **Note** Interfaces to printers are through DOS PRT, COM1-COM-8 ,or LPT1-LPT3. Network printers are only supported through DOS/Net redirection of the COM or LPT networked printers.

12.1.1 Printing Foreign Language Characters

Starting with version 2.4, Modsoft 2.4 has the ability for printing extended ASCII characters from foreign keyboards on certain ASCII printers,e.g. Printronix P300 and P600; and on flat file ascii. This improvement involves creation of a new font that is attached to the flat file and the Printronix drivers.

In effect, the new font simply does not have overstrike characters. It is therefore impossible to print umlat and accent characters. Such extended ascii characters are replaced by their simple ascii character. For example, “a umlat” is simply “a”, “e grave” is simply “e”. This technique produces a listing that is aligned and tabbed properly, if not grammatically correct in the foreign language.

If you already have a printers database “modsoft.pdb” and do not wish to replace it with the distribution copy from “Update”, It may be necessary for you to use the UPF editor to delete flat file print devices you already have, and re-enter them to get the new font installed from the printers.pdb. Be sure you “update” printers.pdb file even if you don’t update the modsoft.pdb.

If you already have printronix P300 or P600 devices in your modsoft.pdb file, it is necessary for you to delete them via the UPF editor and to re enter them.

12.2 Setting Up a Printer

12.2.1 Modsoft Update Considerations

The Printers.pdb file is an option when you update Modsoft. In some cases, the printers.pdb database may have been changed and one or more of your printer types in Modsoft.pdb do not match the main database. If you attempt to edit your Modsoft preference for such a printer, you will get an error message. You should then delete the outdated printer from your list and re-enter it using the UPF Editor.

12.2.2 Starting the UPF Editor

You start the User Preference File editor (independent of Modsoft) by typing:
UPF with the path set to the Modsoft directory. I.E.:

C:\Modsoft> UPF↵

The editor main screen displays a number of user preference related selections including `Print`. To select Print from the keyboard Press the <**Alt**> key to highlight the menu quick keys which for print is < **r** >. Now press < **r** > to display the print pulldown as illustrated. Selectable from the pulldown are; the `Printers` and `Options` functions. These functions can be initiated with the quick keys <**Ctrl + P**> and <**Ctrl + I**> respectively. The menu shorthand keys are highlighted and are selected by pressing the Key. Remember, you can also use a mouse to select these items.

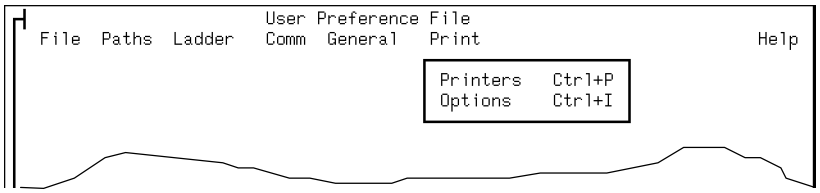


Figure 276 UPF Editor with Print Pulldown

12.2.3 Printers

The Printers selection provides an interface to a data base of printer parameters.
You are provided with a display template and Functional Operation selections on the bottom of the template.

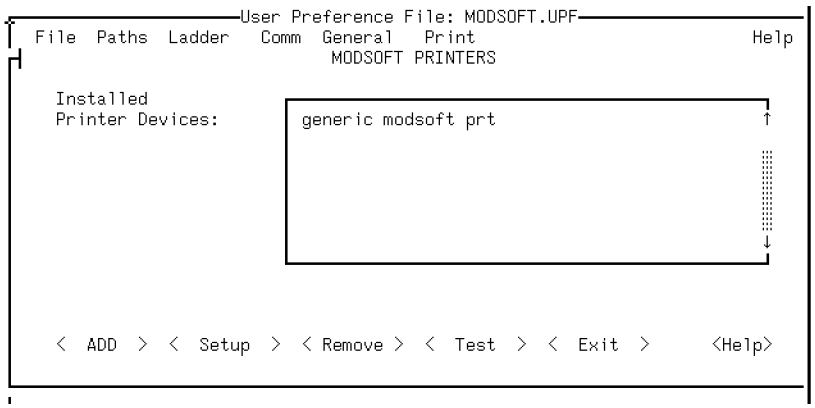


Figure 277 User Interface For Printers Selected

The default Printer file is “Modsoft.prt” in the current directory. It is automatically selected. The Modsoft Print utility uses this file to list printer selections in the Print utility in the first step of customizing your printer.

The Installed Printers (if any) are displayed with the name assigned to them. With the printer name selected you may press the <**Return**> key (or click on the `Setting` selection) to review and edit the existing Setup parameters or remove the Printer from the printer list. Additional user options include Adding a printer or Testing a printer setup.

> ☞ **Note** Printers are interfaced through DOS PRT COM1-8 or LPT 1-3. Network printers are only supported through DOS/Net re_direction of LPT or COM to the networked printer.

12.2.3.1 Add

If you choose the `Add` a printer function < **Alt A** >, the Printer Setup file is displayed. The set-up file fields are filled with default values that you edit to the specifications for the printer you are using. (See the example below).

12.2.3.2 Setup

This selection < **Alt S** > results in the display of a parametric template that you edit to change printing parameters or establish them using the Add Selection.

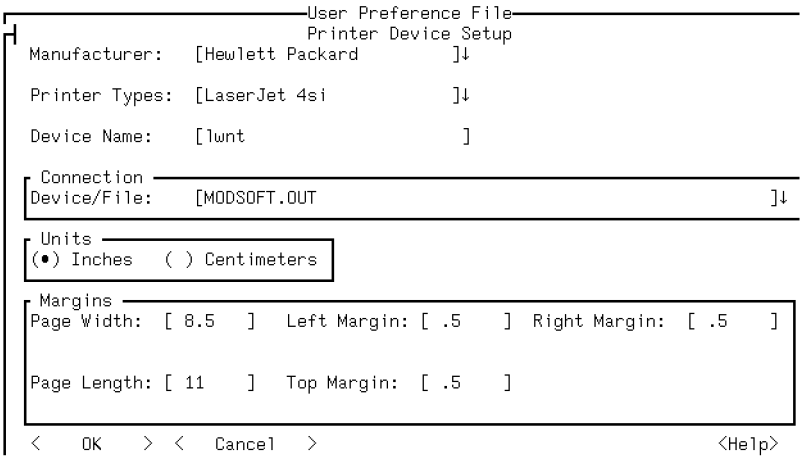


Figure 278 Setup Parameter Example for a Hewlett Packard Printer

12.2.4 UPF Printer Editor Setup

The Setup Parameter screen appears after selecting either Add or Setup. If ADD was invoked, the fields are all set to default values.

Your first selection should be the printer `Manufacturer`, `Type` then the printer `Device Name` that you will be using. The cursor is initially placed in the Manufacturer field and you may scroll through the data base one at a time using the keyboard **up and down arrows** or select a short. scrollable list by pressing the
<**Alt** ↓ > keys. The data base look-up also reacts to entering an *alpha* key, so, if you know the printer you want, you can call it quickly. To leave the manufacturer field type <**Tab**> which moves the cursor down one field. A <**Shift + Tab**> moves up one field.

Using the select string you can add setup information to the printer immediately prior to the start of printing. This setup must be of a sort that does not get “reset” by a general Reset command ESC-E for example. The Printer driver from the database frequently sends a RESET command after processing your selection. In particular, laser printers are handled in this

way. An example of losing your setup occurs with a duplex laser printer being driven with a simplex driver, and a select string to set the duplex feature. This will not work.

Note There is a -Special category for the Manufacturer, which is not a particular company's product. Here, are the generic printer types. These work on most impact printers. Unfortunately, they do not work in Modsoft, on laser printers. For laser printers, set the correct `Manufacturer` and Printer `Device Name` selections.

Expert You can use the `-Special` with Type selections of `Flat`, etc., to ***display*** the printer output file using a text editor. Once established this setup will remain a selectable printer type.

Next, select the DOS device or file name that you want to have most of the time when printing from Modsoft. For example, select LPT1 for the printer connection. You may change your mind at the time of preparing to print from within Modsoft. If you select a filename, enter the name or accept the default in the name box.

After selecting the printer Device Name, the settings of page dimensions and margins and offsets are needed if not already correctly filled in by the data base retrieval. Many impact printers will work with the default values for the margins and offsets. For your laser printer, you will probably want narrower margins and non-zero offsets. The best way to tell how your settings will work, is to TEST the printer selection in the process of exiting the UPF editor. The following is a typical set of data for an HP Laserjet in LandScape mode:

Page	Margins
Width: [11]	Left: [.25] Top: [.4]

Length: [8.5] Right: [.25]

☞ **Note** The Printers.pdb database has been modified to add new printer types in the -Special printer types. This work has modified thename of generic printer types and flat file types. If your existing modsoft.pdb has used these types, it will be necessaryfor you to delete them from your modsoft.pdb file using the UPF editor, and re-enter them.

When you bring up a printer in the UPF editor to ”setup”, and see immediately a yellow stripe that tells you to cancel immediately this indicates that the printers.pdb has changed for the printer type that was selected. It is necessary to follow directions cancel the setup, delete the printer device that has this problem and ”add” it to the list again.We are sorry for the extra effort that you may have to make, but we have improved the overall utility and handling capability of the printers database in this process.

If listings of Ladder diagrams from Modsoft are misaligned, such as the vertical connections don’t line up etc., return to the Setup and check the Margin settings. Then re-test the setup and try Modsoft again.

The large number of devices, available in the database, in all likelihood will satisfy your needs for printing Modsoft listings. Sometimes you will find impact printers, for which the multi-font printing approach that Modsoft attempts to use will noticeably slow down your printing. If this is a problem, you may wish to use one of the “generic” printers under -Special. These Printers are arranged as to the number of columns and lines they produce per page, for example 80 columns x 55 lines. All of the generic printers are single font only.

Modicon has not tested every printer in the Modsoft database. If your printer does not perform correctly, after you have adjusted the margins and offsets correctly, you should revert to using the -Special generic printers.

Some printers in the large Printers.pdb database have multiple fonts but are not remotely selectable. Modsoft may attempt to use one of these fonts without knowing that it cannot be remotely selected. If this happens, the report may be mis-aligned in the ladder lister and other sections may attempt to overprint or just run off the right side of the page. In this case, it is probably best to use the Generic printer types. See the -Special set of printer types list under the manufacturer’s list.

12.2.5 Printer Data Base Detail

Given the above example as an overview, you can review and edit the assignment of printer parameters. The fields are:

Manufacturer The example is Hewlett Packard but the field can be scrolled to select from:

Apple	Antex	Anadex	AmdekAlps
Alphacom	Adobe	Azonics	
Blue Chip	BMC	Brother	
C Itoh	Cal-Abco	Canon	Cardco
Centronics	Citizen	Copal	
Dataproducts	Datasouth	DEC	Diablo
Diconix	Dynax		
Epson	Fujitso		
Hewlett Packard			
Genicom			
IBM	Infoscribe	Integral Data Systems	
JDL			
Juki			
Kyocera			
LaserMaster			
Mannesman	Talley Matra	Micro Peripherals	
N A Qantex	NEC		
Okidata	Olivetti	Olympia	
Panasonic	Personel Micro Comp	Primages	Printronix
Qms	Quadram	Qume	
Ricoh			

Sakata	Sanyo	Seikosha	Shinwa
Silver-Reed	Smith-Corona	Star	Sumitronics
Tandy	Texas Instruments	Toshiba	Transtar

-Special

Printer Types Refer to the Manufacturers Printer Documentation and match the Type by scrolling through the entries in the type field or press <Alt ↓ > to display the selection list which for the Hewlett Packard example includes:

Deskjet
Deskjet Plus
Deskjet Plus (LS) for Landscape
Deskjet Portable
Deskjet Portable L
Deskjet 500
Deskjet 500 (LS)
Deskjet 500C

☞ **Note** The above ***types*** are specific to the HP used as an example. Actual types will vary depending on the Printer Manufacturer you select.

Device/File Select the name you want to assign to the Device or File The selection list includes:

Modsoft.Out
LPT1
LPT2
LPT3
COM1
COM2
COM3
COM4
PRN

There are additional setup fields comprising *Page* dimensions, *Margin* requirements. The *Unit* of measure (i.e., Inches or Centimeters) is not a setup field. You can select the unit most convenient for you while editing the Page and Margin fields..

When you are satisfied with the selected parameters you can save them to the Modsoft Printer data base (Modsoft.pdb by keying the OK <**Enter**> If no changes have been made or you do not want to save what changes have been made, key the cancel function <**Alt** +**C**>.

12.2.6 Remove

Whenever you wish to `Remove` the specification of a Printer from the installed list, just select the Printer name from the list and key <**Alt** + **R**>

12.2.6.1 Test

When establishing a new printer setup or editing an old one, the UPF editor provides a test function to verify the correctness of the parameters you set. Key the sequence <Alt + T> with the listed name selected and you will be advised that the UPF editor is Exited to perform the Test routine. Given the example setup, a file called Modsoft.out with test parameters will be created. You can “Copy” it to your Printer at your convenience.

☞ **Note** Test can return a message referring you to the “Slate” software. Since this software is embedded in the Modsoft User Preference file you are being directed to enter the UPF editor and review the printer specifications for error or omission.

12.2.6.2 Exit

This Selection will close out of the printers function and return the UPF Editor to the initial entry screen.

12.2.6.3 Help

This selection provides a brief summary of the database edits that are enabled using this Display.

12.3 Options

This is the second entry under the Main menu Print pulldown. When selected a screen is displayed.

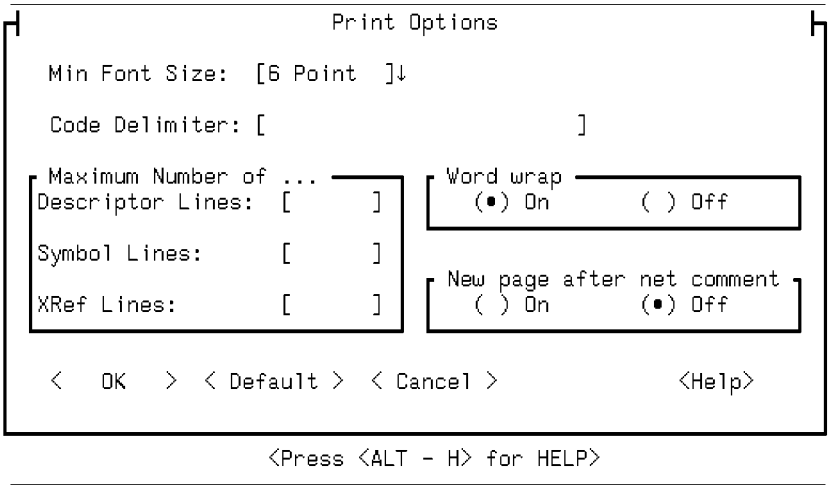


Figure 279 Print OPTIONS Display

The Print Options template provides the entries necessary to stylize the listing output. All fields requiring initial data contain default values which can be accepted by keying the sequence <**Alt** + **O**> . During an edit you can return to Default values by keying <**Alt** + **D**> . Variable data can be selected by scrolling through the field of interest including:

Min Font Size 6 Point

CodeDelimiter This is a short, unique string that is used for tagging a command sequence within reference comments that are to be exported to an MMI such as Unicel. If not using this way leave the field blank.

MaximumNumber Of ... This box allows you to set the Maximum number of :
Descriptor Lines
Symbol Lines
Xref (Cross-reference) Lines

In addition, **Word Wrap** can be toggled between On and Off. A similar provision is made to start a **New pageafter a network comment.**The default for word wrap is **On** and New Page is **Off.** The Accept (OK) and Cancel are the same as previously documented.

12.4 Modsoft Print Function

The Modsoft documentation package extracts information concerning the application and formats it in various operator specified ways. The resultant data can be printed to a local printer or stored on a disk file. The documentation package is entered by selecting the item "Tools" "Print" (from the Main Menu illustrated.

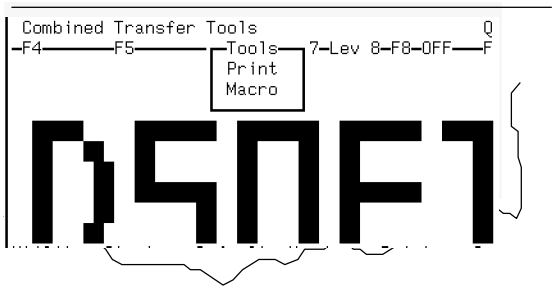


Figure 280 The Main Menu TOOLS Pulldown

After you select `Print`, the program source file select list for documentation is displayed as follows:

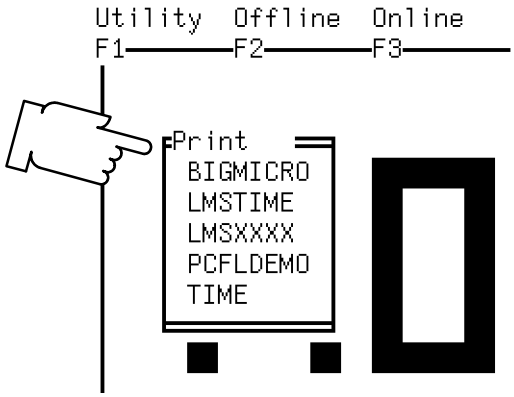


Figure 281 Example of Print Function File select Screen

If you have set the proper environment for printing the "Time" example, the parameters can be filled out or the standard defaults can be used. How you select the options provided on the Print Menu line and its pulldowns, depends on your needs.

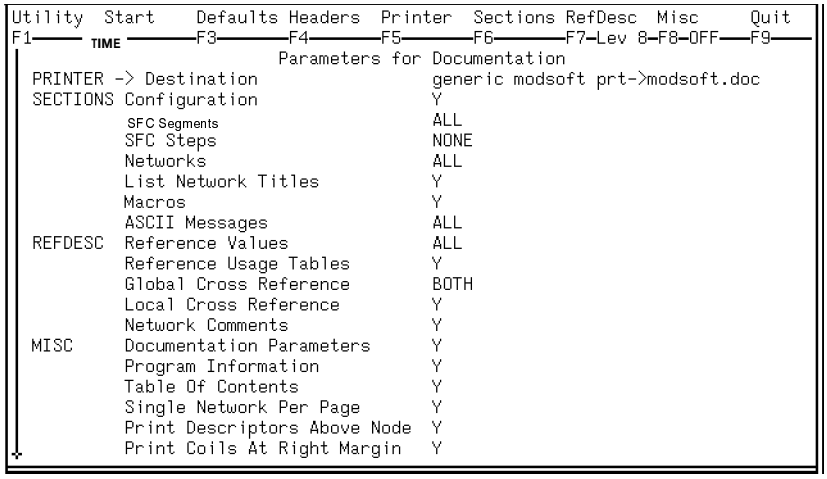


Figure 282 Print Menu Initial Entry screen

If you have completed a project, you will want the maximum amount of documentation printed for your archive file. However, if you are in a development mode, you can select only the program elements you are most interested in.

> ☞ **Note** *If you are running the Modsoft application with the switch with SFC disabled, no SFC references will be visible in a listing.*

When a new program is first displayed, the user-selectable parameters are set to default values. You may leave the selections as they are, by selecting Start directly. Each of the available user chosen parameters are described in the following paragraphs and applies to the software program set in the Panel.

The Equation Network prints out simply as a screen image, without further elaboration.

Printing of new math networks will be supported as a new network type in the documentor. The Print operations will print from the identical screen window as the ladder editor provides. Printing of math networks does not require any additional user parameter selections. New math screen dump therefore does not have symbols and descriptors. Math networks will print local xrefs of the coils. In general, global xrefs will also print the math references in the global xref list.

12.5 Print Parameters

12.5.1 Printer

12.5.1.1 Print Direct

If you use the DOS convention “LPT1”, the listing produced will be directed to the Printer attached as LPT1. No file will be created on the disk drive.

12.5.1.2 Print From File

Modsoft listing files are in special formats, as though the data had gone directly to the Printer. To Print the listing, send the file to the printer directly, using “COPY” file to LPTx. Do not use Printer file formatting facilities.

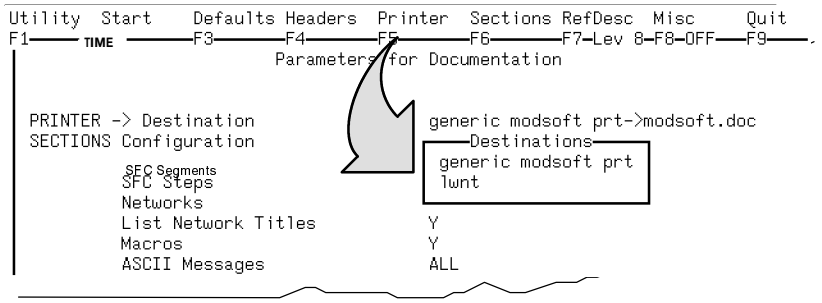


Figure 283 Printer Parameters Selection

12.5.1.3 Destination Select

The list box is displayed to allow your selection from printers you have defined. In the illustration above, for example, the “lwnt” destination and the generic Modsoft.Prt printer have previously been defined. The definition includes all formatting information i.e. page size, margin, type font etc,.

If in building the list of destinations, a destination with bad data for the device is encountered, you will see a system message to that effect, naming the bad device. In addition, the position in the table which had the bad device will also indicate the bad device. If you have forgotten the device name that was bad, it can be seen by selecting the “bad” entry. Be sure not to try to print to the bad device. The most common cause for bad device data is missing font files for printers requiring down-load fonts. Be sure that your installation of the UPF editor is complete in placing the font files in the proper place (modsoft\runtime\fonts\...).

Tip: The destinationfile or DOS device name *in the last existing program's .ENV file* is the initial destination for the printer.

12.5.1.4 Device or Filename select

You may retype the file or device name to change it. For example, if the database default is modsoft.doc this will be displayed. You may change this to any legal file name or DOS device name. If selecting a DOS device, do not append a colon, eg. LPT1: is illegal. The system will not let you complete the entry. (See your DOS manual for legal printer device names.) Also note, your DOS assignment of print ports must be done prior to using modsoft, usually in your autoexec.bat.

Note If you attempt to Print to a non-existent port you may receive an error such as:

Unable to open parallel printer Port # 15

12.6 Defaults

12.6.1 Default Parameters

During the option selection process, you are allowed to restore the base defaults by selecting this Parameter menu entry.

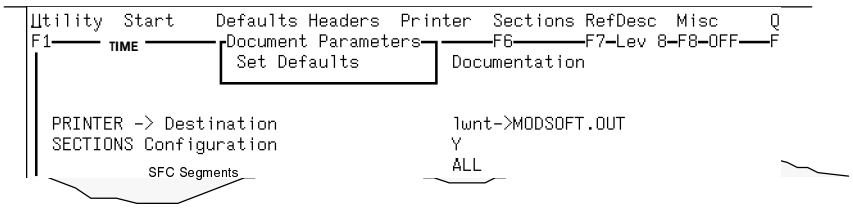


Figure 284 Initialized Default Value Display

The defaults are the initial entries in the variable fields of the parameter documentation screen. This selection is provided to allow you to return to a known set of parameters from which you can customize a new listing. Returning to this screen from another execution of Modsoft you will find the last known field values rather than the initial default.

12.7 Headers

12.7.1 Header Filename

When the cursor is on this field, you may specify the absolute path name of a user created ASCII text file (example c:\modsoft\PROGRAMS\x.HDR) in which you have constructed a general description of the program about to be documented. The contents of this ASCII file will be printed out once starting on page 1 of the listing.

12.7.2 Page Header and Footer

When the cursor is on the header or footer field, you can enter up to three lines of ASCII text, which is printed on the top or bottom of every page in the listing. You can specify the header or footer by typing in @FILENAME where the file given by filename contains the ASCII text for the header or footer. The maximum combined size for the header and footer is 10 lines If you use the @ Filename convention otherwise the Header/Footer has a maximum of 3 lines each. If you specify files whose combined contents exceed this limit, then only the first 10 lines of the combined ASCII texts are printed.

12.7.3 Meta Commands

Headers and footers can be tailored to meet your documentation requirements, by formatting with Meta commands.

Expert When a header or footer is specified by a control file or by entering the header and/or footer in the user interface dialog, the default head and foot lines are not used; thus *time, date* and *pagination* features are **not** automatically provided.

12.7.3.1 Meta Command Summary

The Meta commands provide positioning and data functions in two categories:

- Position Commands

  <CENTER> and <RIGHT>

- Print Data

  <DATE>, <TIME>, <PAGE>, <PROGRAM>, <NET>, <SECTION>

Literal strings are differentiated by enclosure in quotes as in "<DATE>" or simply typed as text without the <> delimiters i.e. Date.

<DATE>	Provides the system date as derived from the Controller.
<TIME>	Provides the system time as derived from the Controller.
<PAGE>	Provides a running page count
<PLC>	Documents the currently configured PLC
<PROGRAM>	Name of the current program
<NET>	The number of the network being printed
<SECTION>	This labels the listing in categories i.e, "Configuration" "Macro", etc.
<CENTER>	Centers all following text and evaluated mete-characters, up to the end of the line or until a <RIGHT> command is encountered.
<RIGHT>	Right justifies all following text and evaluated mete-characters, up to the end of the line.
" "	Quoted text is literal. "<DATE>" is = <DATE>

12.7.3.2 Meta Command Example

If you create a combination of meta commands and literal text like:

<DATE> <CENTER><PROGRAM> <RIGHT>PAGE:<PAGE>
<CENTER>"SEGMENT:"<SECTION> NETWORK:<NET> <RIGHT><TIME>

You will see a listing with the printed characteristics of:

03/23/95	APROGRAM.ENV	PAGE:13
	SEGMENT:02 NETWORK:05	11:25:00

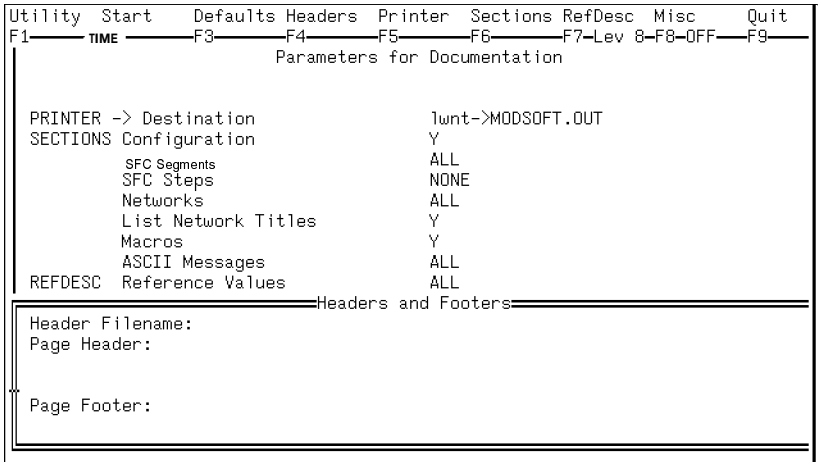


Figure 285 HeaderFilename and Footer Text Entry

12.8 Sections

Using the parameter select menu, you can customize the contents of the listing. The Section selection allows the choices.

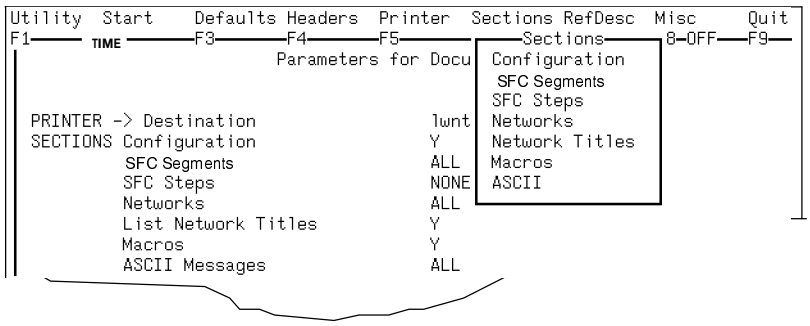


Figure 286 Section Custom Selection

12.8.1 Listing Annotation

Some of the information generated by the “Print” is a summary of configuration information and Program Logic Symbols available on the programming panel, while other information is a result of Print compilation. When you have created a listing, additional “Print” information is provided as a tag to Input/Output references. This tag information is in the form: Absolute Network and Segment information

Logic summary and various cross reference listing types also produce additional information about node location and use. You will see reference information in the generalized form:

#001r2,1 -]/[#001w1,* -( )

where:
The Network Number is followed by whether the element is a Read or a Write type, its Row and Column location and its instruction type. In the above example, the read type is a normally closed contact located at row 2, column 1 of the network. The write is a similarly referenced coil on row 1. The * is used in the case of coils, because although they are listed in the right-

most column, they may appear in any column position in the network as long as it is the last logic element used in that row.

☞ **Note** The above format represents the SFC OFF condition. With SFC ON the reference is a Segment Object Network, where the Network is relative within the Object.

12.8.2 Configuration

When the cursor is on this field, you may choose whether or not (Y,N) the listing includes the configuration details of the target processor for which the program was developed. The default is to ***print*** it. These configuration details are listed in the following order:

Configuration Overview
I/O Map
Ports
Segments
Loadable
Configuration Extension Blocks

- Configuration Overview

This data is printed in the same format as entered when you established the configuration by using the “Offline” Configuration menu.

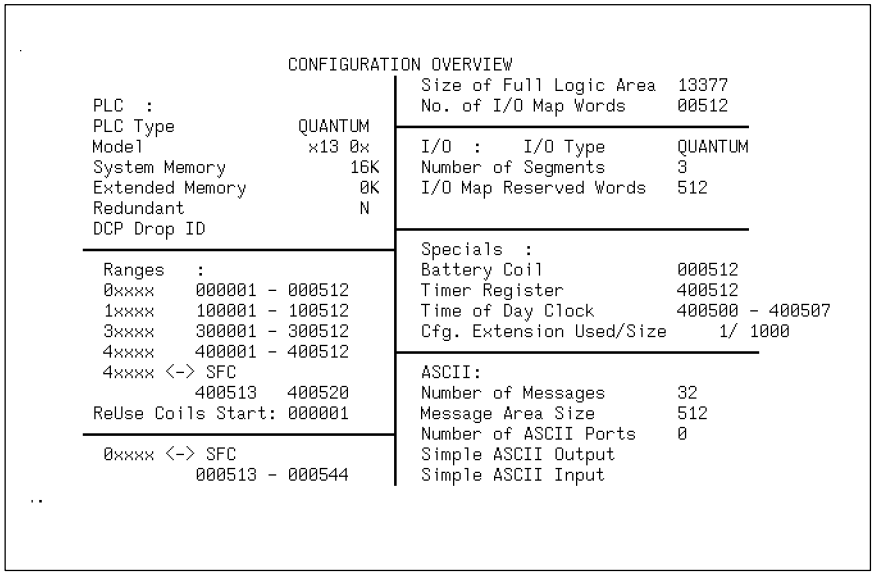


Figure 287 Example of Configuration Listing

I/O Map

Each rack has the following data listed

Rack Number
Number of Input Bits defined
Number of Output Bits defined
Module Type
Input Reference, if any
Output Reference, if any
Data Type
Module Description

The I/O Map is listed with two racks shown on each page. The general format of the I/O Map Listing is illustrated. The Listing will skip all empty racks.

12.8.2.1 Quantum Specific Configuration

The documentation of module parameters Zoom was added to the Lister in Modsoft 2.2 (and above), immediately following the printing of each drop's IOMap.

The selection is elected at the Configuration select prompt where a list box allows you to pick from:

Configuration Y ->

Y
N
W/ModZoom

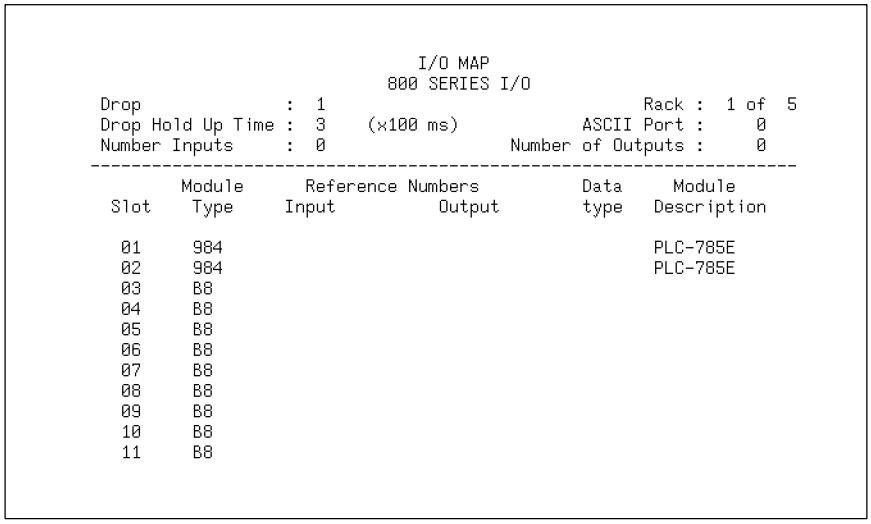
Where, **W/ModZoom** would be active only for Quantum PLC's. This keeps the old .env info equivalent as far as the Y/N are concerned. The **W/ModZoom** selection means **Yes** to the configuration and also to add printing of the quantum module parameters.

An example of a module zoom parameter printout is shown below:

The number of non-empty lines of each printable screen will determine the page breaks. If it is too long to fit on the current page, a page break will be inserted.

```
              = = = = = = = = = = = = = = = = = = = = = = = = =
                         140 DDO 353 00: 32-OUT 30VDC            Screen 1
Head-Slot: 0  Drop: 1  Slot: 3
                                          Output Type:     BIN
                                          Timeout State:   User Defined
   User Defined Timeout State Points 1-16:     0000000000000000
   User Defined Timeout State Points 17-32:    0000000000000000
                             End of DDO 353 00 Zoom
              = = = = = = = = = = = = = = = = = = = = = = = = =
```

Figure 288 Quantum Zoom Parameter



```
                              I/O MAP
                          800 SERIES I/O
Drop              :  1                                  Rack :  1 of  5
Drop Hold Up Time :  3    (x100 ms)            ASCII Port :     0
Number Inputs     :  0                  Number of Outputs :     0
-------------------------------------------------------------------------
         Module      Reference Numbers          Data     Module
 Slot     Type     Input          Output        type   Description

  01      984                                          PLC-785E
  02      984                                          PLC-785E
  03      B8
  04      B8
  05      B8
  06      B8
  07      B8
  08      B8
  09      B8
  10      B8
  11      B8
```

Figure 289 Example Of Data Format For I/O Map Listing

The `Loadable, Config. Ext` and `Port` data are listed as available.

- Loadables
  This is a list of the loadable DX functions and user loadables that exist in the program.

- Ports
  The Communications Port parameters that were set or are at default settings are printed in this listing section.

- Configuration Extension Blocks
  Configuration Extension data is documented in this area.
  If there is a VME486 Config Extension it is appended to the listing in the same format as it appears on the display

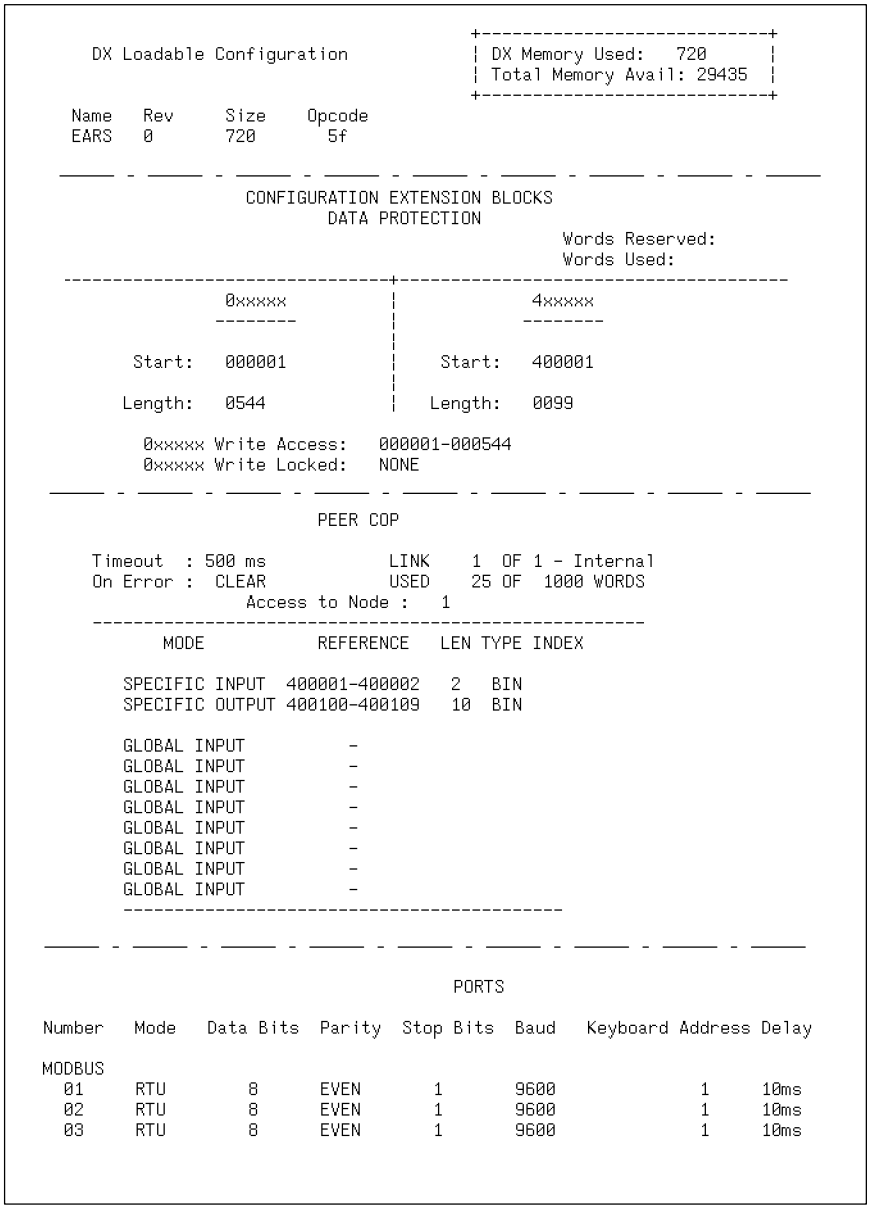


Figure 290 Loadable, Configuration Extension, Peer Cop and Port Configuration Data

Segment Scheduler

This information is listed for your information. It is identical to the screen display in the Configuration editor.

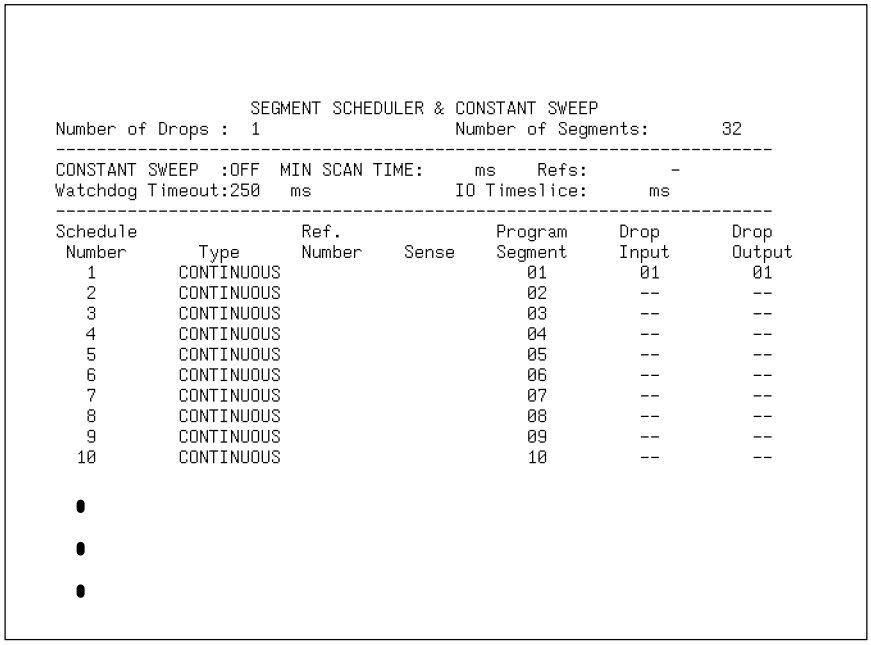


Figure 291 Segment Information

12.8.3 SFC Segments

You can select from the selection list:

ALL
NONE means no segment or network will be printed
SELECTED 1, 2, ...32 for example

Thus you have control over the entire range of choices for documenting segments.

☞ **Note** In non SFC mode, the SFC Segments selection is set to NONE, and you can not change it. The menu item is “Greyed”.

When in SFC each of the three selections are valid. If you choose ALL you will get all unconditionally. Within SFC what you select for segments is what is printed.

☞ **Note** Ladder Logic within any segments not enabled will not be printed.

12.8.3.1 SFC Steps

When the cursor is on this field, you may direct the Listing (by selecting All, None or Selected) to have the SFC representation of the program included in the listing, or you may suppress the listing. If SFC is selected, then a logic diagram for each selected segment showing the steps and transitions of the program and their interconnections are included in the list file. If you select None, no SFC structure is printed.

The following diagram is a listing of the SFC elements used in the segment together with their annotation.

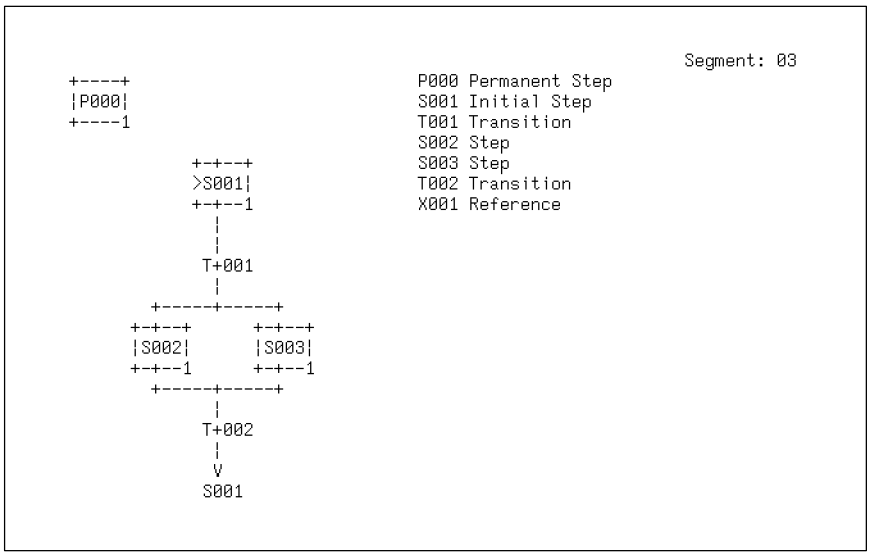


Figure 292 Listing Example of SFC

Selected SFC Elements

If you have selected **exactlyone segment** (see above) to be included in the listing, the user may direct the Print to restrict the listing to only those SFC elements whose identification numbers are entered. The elements are entered by keying in their numbers and separating them with commas, or they can be indicated by ranges (e.g., “S1-S9”,T3-T7). If you make no selection in this field, or if you type in the letter “a”, then all of the SFC elements are selected to be listed.

12.8.4 Networks

When the cursor is on this field, you may select from:

ALL
NONE
SELECTED

These selections allow you to have the Print include a listing of all the networks in each segment together with any associated network name or None, or selected Networks only. A Network from the Time example when printed on a printer in portrait mode would look like:

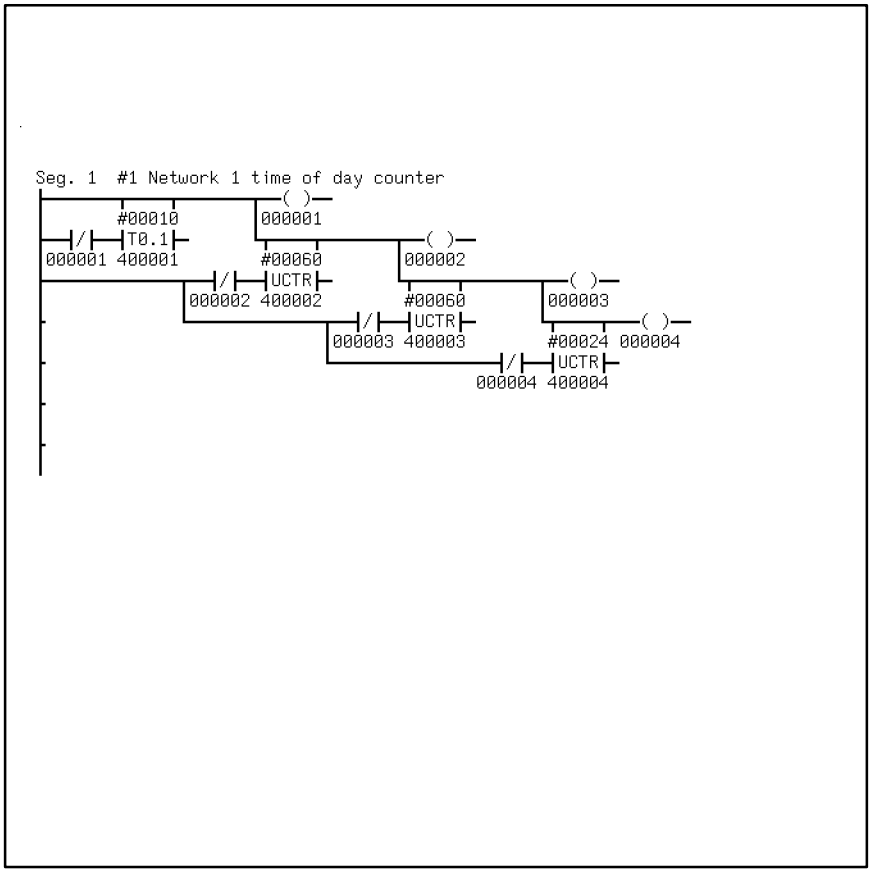


Figure 293 Portrait Mode Printing

12.8.4.1 Listing Element Specifics for Ladder Diagrams

1 Bottom character of power rail: That the network listing is complete in 5 rungs and does not extend to the next page can be seen from the _|_ at the bottom of the power rail. If the network would be continued on the next page, the bottom of the power rail would simply show | .

2 Reference Comments are being printed above the node and symbols therefore are below. The user preference file has been used to limit the number of comment lines to 3 and the number of symbol lines to 1.

3 Font Sizes: Where possible with the destination printer, the comments and symbols and cross references on the network listing are in a small font to conserve paper, and make the listings more legible. The rung graphics are printed in the larger font size and font style for clarity.

4 Cross references: When paper width permits, the local coil xrefs, are printed to the right of the coils. If more lines of xrefs are needed than will fit next to the rung, then the word (more) will appear below the last xref printed. If the xrefs won't fit to the right, then local xrefs are printed on the bottom of the page. In this case, the listing of xrefs includes also the comments and symbols. The nomenclature of xrefs is different in flat structure than in SFC structure. In FLAT, you will see NC:#0001, which simply means that the xref is to a normally closed contact in absolute network #0001. In SFC, you might see NO:03S001#0001 this is interpreted as Normally open: Segment 3; Sequential Step 003; network #0001. Variations on these are as follows:

Contact Types: NO: normally open
NC: normally closed
PT: positive transition
NT: negative transition
CO: coil
CL: coil (memorized)

MC: Macro reference(no prefix) registerStep types:P permanent S sequential T Transitional

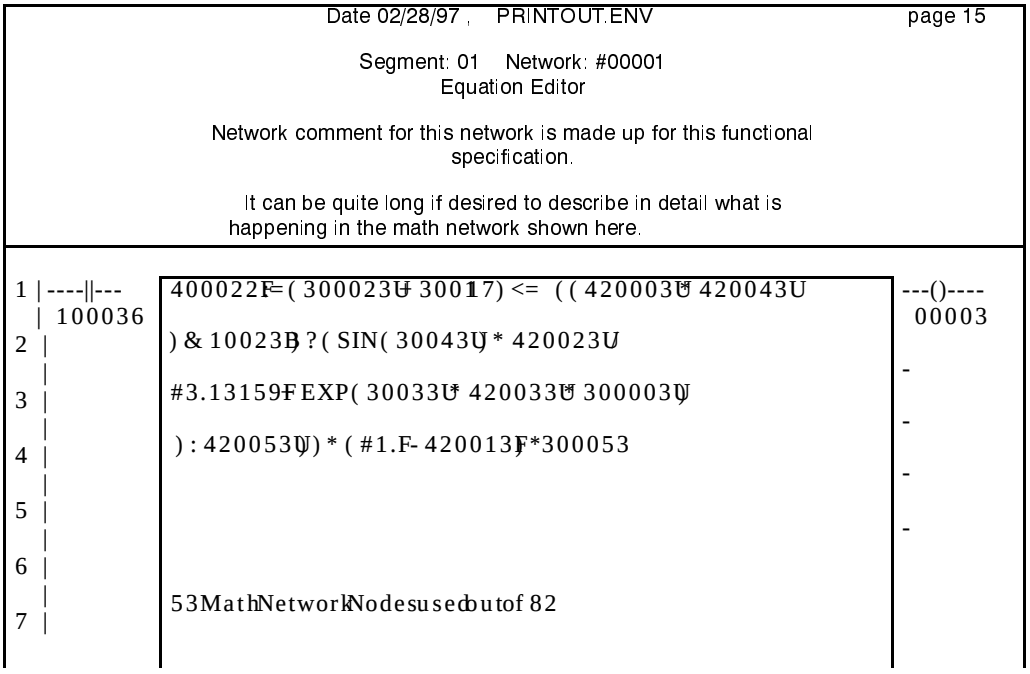
5 Point of origin:

This is the info on the lowest line below the reference node. It looks like {1.1} in the flat listing and means (in this case) the coil is in network 1, rung 1. In the case of SFC networks you would see {1P000 1.1} or {2S002 1.4} or {1T001 1.3}.

In these cases, the prefix number 1,2, etc give the segment number in which the coil is found. The P S and T mean respectively, Permanent Step, Sequence Step and Transition step. The next value 001 is the step number and the 1.1 is the network and row as before.

12.8.4.2 Equation Network Printing

The Print layout has the information that appears in the regular (non-wide) mode display plus, in a table at the bottom, there is a local cross references (if selected by user). Global Cross references for all references used in this network are produced if selected by the user. The layout below will fit in portrait mode on 8.5 inch by 11 inch paper. On wider paper, the format does not change.



12.8.5 Network Titles

When the cursor is on this field, you may answer Yes or No to have Print include a listing of all the networks in each segment together with any associated network name.

12.8.6 Macros

Listing of Macro expansions selected from a `Yes` or `No` choice:

If you select Yes, you create a Network expansion of the Macro implemented via the data in the Macro call blocks. You may enter No to exclude listing Macros.

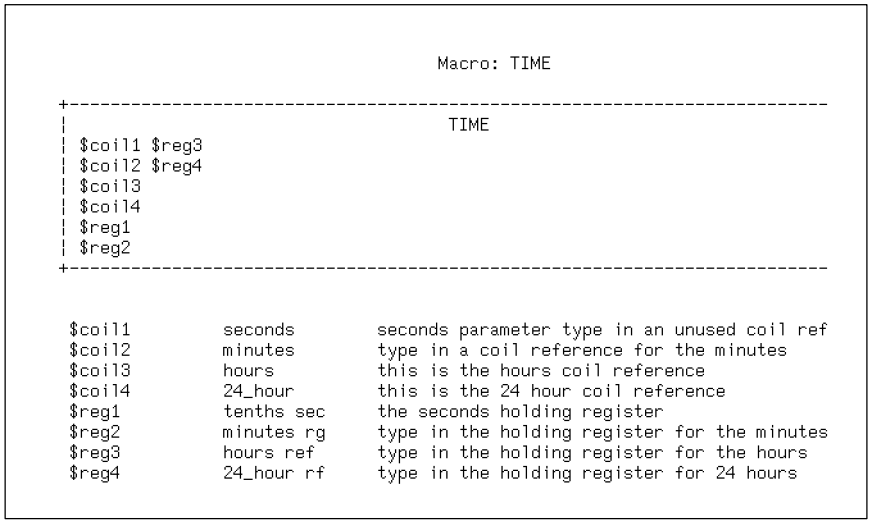


Figure 294 Example of Macro Expansion Diagram

☞ **Note** This feature only expands the Macro that is invoked in Segments/Networks selected for Printing.

12.8.7 ASCII Messages

When the cursor is on this field you can enter a range of "Selected" ASCII message numbers or you can specify "**all**" to select all ASCII messages or " 1..2..3 etc. through **n**". The ASCII message formats for the specified range of messages are listed in consecutive message number order. You can suppress ASCII printing by selecting None. The formats are:

message #1 <message #1 format>
message #2 <message #2 format>
.
message #n <message #n format>.

where *n* is the highest message number specified. If the number of messages to be listed is too large for them to fit on one page, then as many additional pages as are necessary will be used. The illustration below shows a typical ASCII Listing. You may optionally select ***None*** or ***Select*** a range

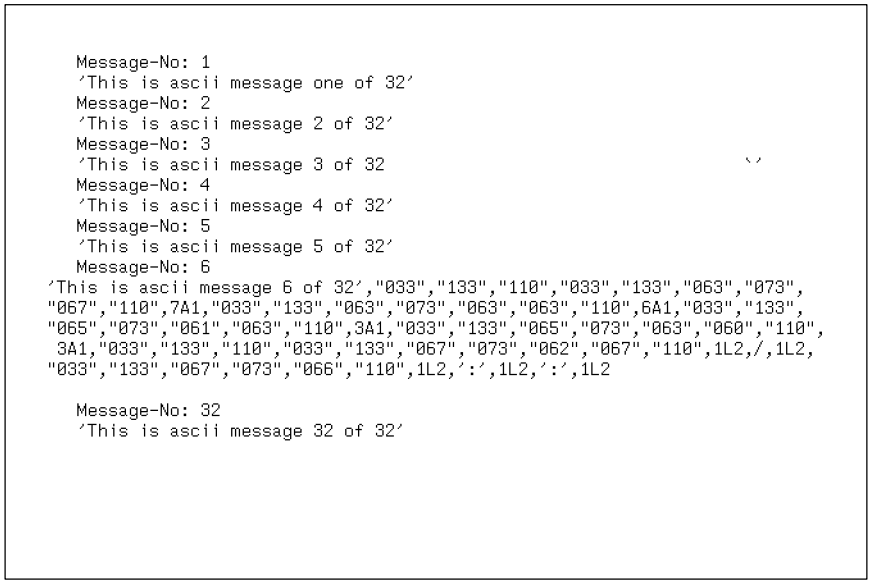


Figure 295 Time Example ASCII Listing

12.9 References and Descriptions

Your listings can contain commentary and cross-reference information in a variety of formats. This menu selection provides the options as illustrated below.

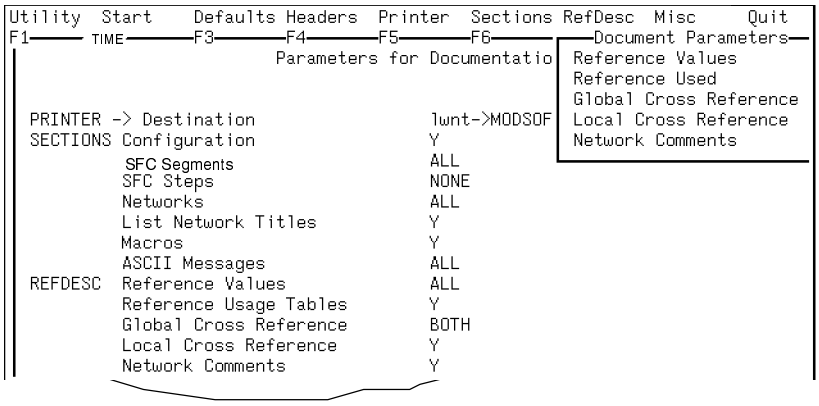


Figure 296 Reference and Description Options

12.9.1 Reference Values

You can select one or more of the reference types and the listing will include a dump of the selected types that have values. You are allowed options:

ALL
NONE
SELECTED

where “All” is all 0x, 1x, 3x, and 4x references and “None” is for no listing. You also can select the mix of reference types that you want to print with the “Selected”. If you use an “N” in the list it is limited to non zero values only.

The default is to print all references. The format provides for 5 digit decimal, 4 digit Hexadecimal or 2 character ASCII data.

6x registers are printed only by designating a “6” in the select list

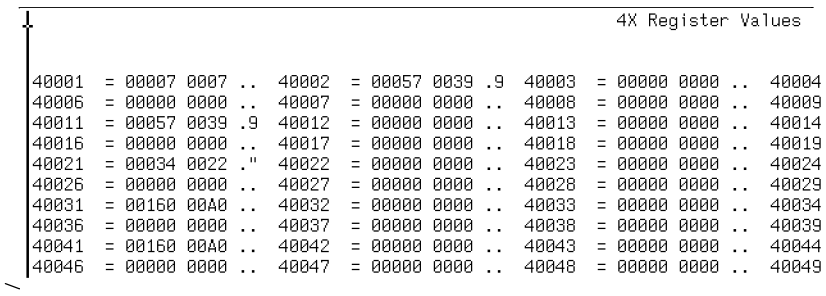


Figure 297 Reference Values

12.9.2 Reference Used

When the cursor is on this field, you can direct the documentation package to list a diagram showing, for each of the reference types, which reference numbers have been used in the program. The 0X Register listing illustrated in is typical of the format used. Your option is to print all “Y” or none “N”. The 0X and 1X listing is printed 48 items to a line. The 3X and 4X are printed 50 to a line.

12.9.2.1 Reference Used Key

The Used reference listings provide a visual aid to help you see what configured references are used and how the references were used. Blank spaces (**Empty**) are not used (i.e. if you look at the configured reference range, these are beyond the range found you could include them by re-configuring). The **Dash** relates to the reference that is in the configured range and is not assigned any value, therefore it is useable in further program development with no configuration change needed.

Starting with the reference listing you can find if a reference is used while defining the I/O Map but has not been programmed in logic yet. This condition is indicated by the **M** in that reference location.

If the reference is used both in the I/O Map AND Logic a **B** is printed in that reference location. An **L** indicates the reference is used in logic only.

If configured for SFC, those registers assigned to the SFC logic are marked with an **S**.

If you have initialized a reference with a symbol, tag or descriptor that document it but have not implemented that reference in Logic a **?** is printed. If that same reference does not have any symbols or commentary to define it, a lower case character is printed.

☞ **Note** This condition also occurs if none of B, M or L are true, the lower case is used to indicate Not Used.

```
   Time: 14:38:26     Date: 03/22/95 TIME785.ENV                    Page: 38
                              Used 4X Registers
              400000   L L L L L L - - - 400010 L L L L - - - - - -
              400020 L L - - - - - - - - 400030 L L - - - - - - - -
              400040 L L - - - - - - - - 400050 - - - - - - - - - -
              400060 - - - - - - - - - - 400070 - - - - - - - - - -
              400080 - - - - - - - - - - 400090 - - - - - - - - - -
              400100 m m m m m m m m m m 400110 - - - - - - - - - -
              400120 - - - - - - - - - - 400130 - - - - - - - - - -
              400140 - - - - - - - - - - 400150 - - - - - - - - - -
              400160 - - - - - - - - - - 400170 - - - - - - - - - -
              400180 - - - - - - - - - - 400190 - - - - - - - - - -
              400200 - - - - - - - - - - 400210 - - - - - - - - - -
              400220 - - - - - - - - - - 400230 - - - - - - - - - -




Key : UPPER CASE LETTER = DOCUMENTED; lower case letter = un-documented
      B = In both Logic and I/O               - = Available
      L = In Logic only                       ? = Documented, but not used
      M = In I/O only                         S = Configured for SFC
```

Figure 298 Reference Usage Example

12.9.3 Global Cross Reference

The Global Cross Reference listing selection allows you to select both Alpha and Numeric lists or select no Listing of this type i.e.,

Both or
None.

☞ **Note:** The print output width is determined by the overall requirement you specify. In these cases xrefs to the right of coils are not able to fit and the word MORE is printed to inform you of that condition. Although not printed they are documented if you have selected global cross reference printing.

12.9.3.1 Listing Nomenclature

Provides for each reference:
Symbol and Comment
Locations where used

Formats:
Ladder or SFC Discretes:
examples NC:#0001 or CO (or CL) :01P000#0001
(contact type:network)
Contact types:
NO: normally open
NC: normally closed
PT: positive transition
NT: negative transition
CO: coil
CL: memorized coil
MC: (if in macro)

network
non-sfc: #nnnn
sfc: SegStepNetwork eg. 01P000#0001

Ladder or SFC Registers:

examples #0002 or 03S002#0003
MC:... (if in macro)

network:
non-sfc: #nnnn
sfc: 03S002#0003

Traffic Cop discretes and registers alike:
example H1_D01_R1_S07_w; (alt _r)
stands for: Head#1,Drop#1,Rack#1,slot#7,write (alt read)

Note:Headnumberis specificto Quantumonly!

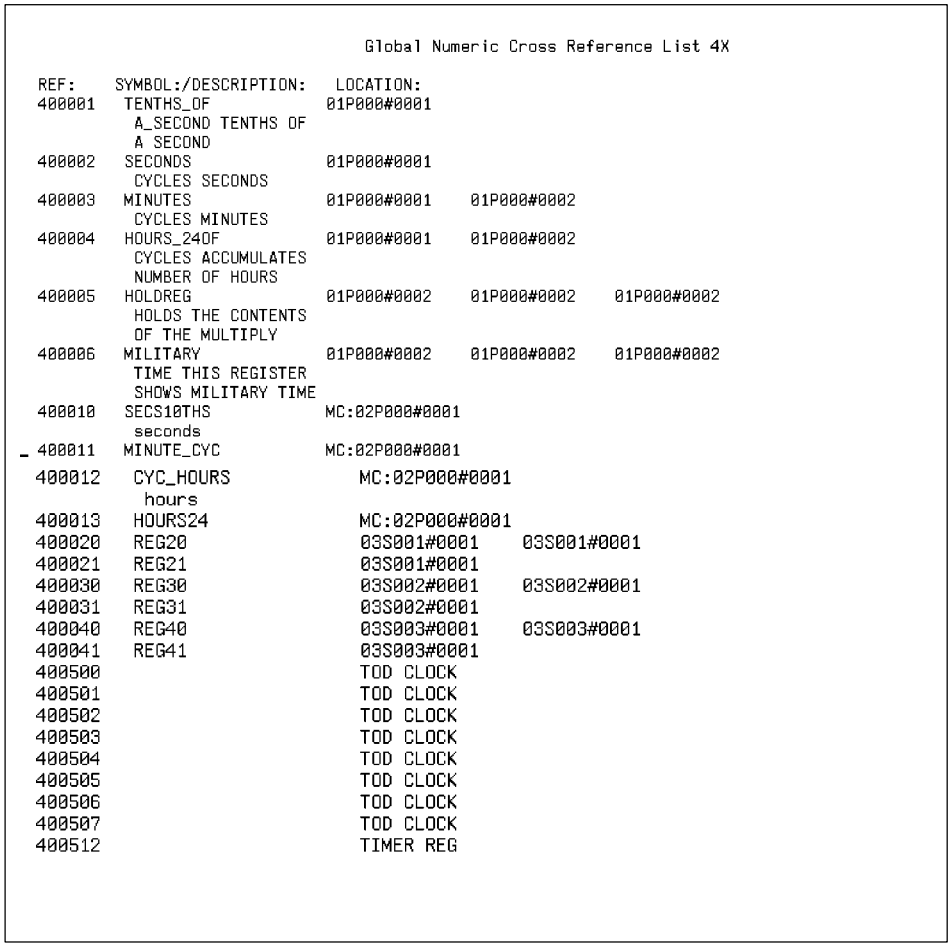
Peer Cop discretes and registers alike:
example 01.SpOut.001 or 02.SpIn.004
01.GlbOut or 02.GlbIn.003

stands for: LINK.TYPE.NODE TYPE: SpOut, SpIn = Specific out/inGlbOut, GlbIn = Global out/in

☞ **Note** GlbOut sends to all nodes and has no need of NODE number.

12.9.3.2 Global Xref Numeric Order

When the cursor is on this field, you can direct the documentation package to compile and list (for each reference) the positions in the logic where the reference is used. You can see in the illustrated listing, that the *USED* data starts with the first two references then can put up to 5 more data sets on successive lines.This listing is in numeric order of the reference numbers. Each reference number is listed together with any associated symbolic data and comment and state or content. Then every location it occurs in is recorded by giving the segment number and the network number within the segment. The global cross reference includes all implied references.

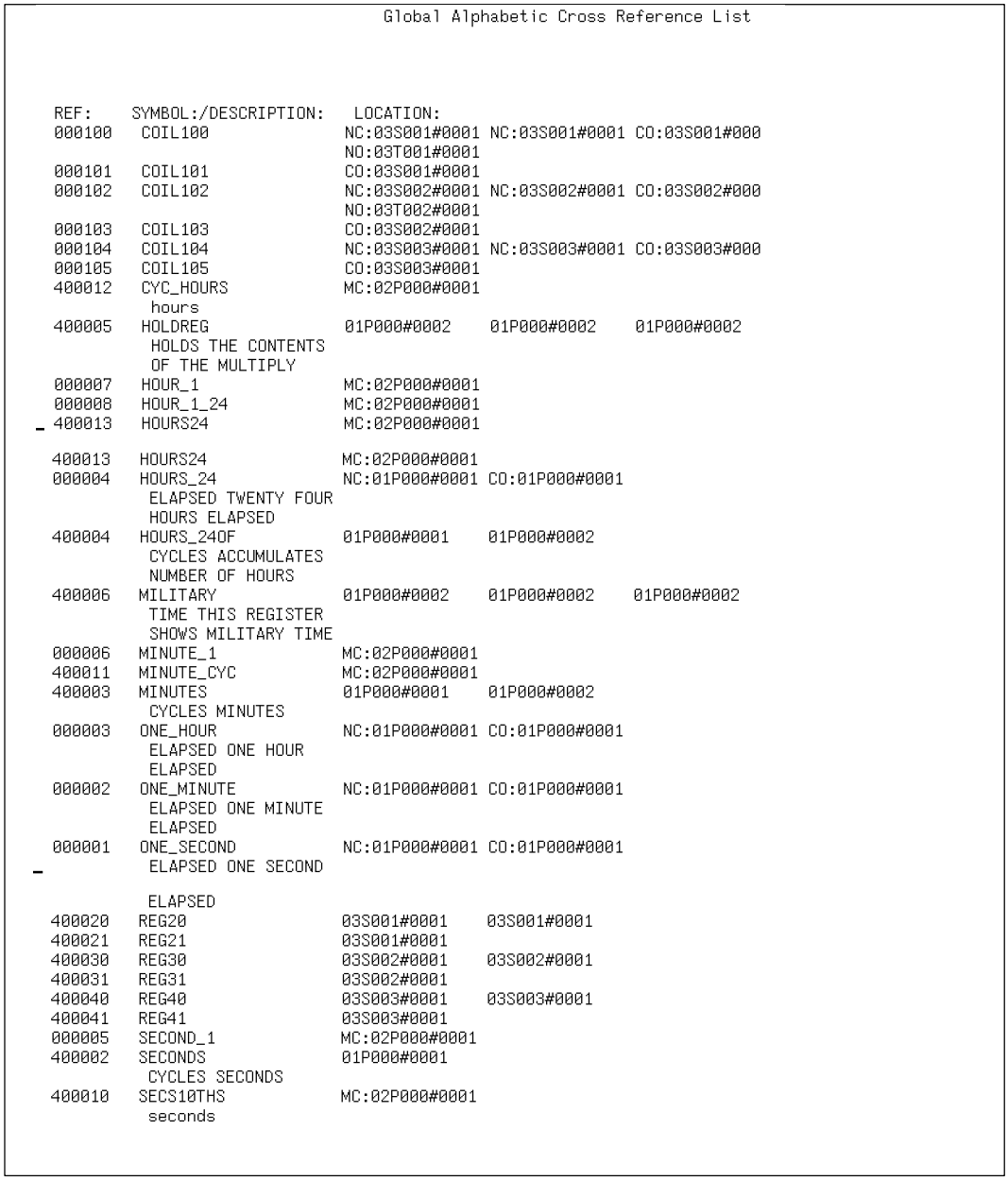


```
                                Global Numeric Cross Reference List 4X

 REF:    SYMBOL:/DESCRIPTION:   LOCATION:
 400001   TENTHS_OF             01P000#0001
           A_SECOND TENTHS OF
           A SECOND
 400002   SECONDS               01P000#0001
           CYCLES SECONDS
 400003   MINUTES               01P000#0001    01P000#0002
           CYCLES MINUTES
 400004   HOURS_24OF            01P000#0001    01P000#0002
           CYCLES ACCUMULATES
           NUMBER OF HOURS
 400005   HOLDREG               01P000#0002    01P000#0002    01P000#0002
           HOLDS THE CONTENTS
           OF THE MULTIPLY
 400006   MILITARY              01P000#0002    01P000#0002    01P000#0002
           TIME THIS REGISTER
           SHOWS MILITARY TIME
 400010   SECS10THS             MC:02P000#0001
           seconds
_400011   MINUTE_CYC            MC:02P000#0001
 400012    CYC_HOURS               MC:02P000#0001
            hours
 400013    HOURS24                 MC:02P000#0001
 400020    REG20                   03S001#0001    03S001#0001
 400021    REG21                   03S001#0001
 400030    REG30                   03S002#0001    03S002#0001
 400031    REG31                   03S002#0001
 400040    REG40                   03S003#0001    03S003#0001
 400041    REG41                   03S003#0001
 400500                            TOD CLOCK
 400501                            TOD CLOCK
 400502                            TOD CLOCK
 400503                            TOD CLOCK
 400504                            TOD CLOCK
 400505                            TOD CLOCK
 400506                            TOD CLOCK
 400507                            TOD CLOCK
 400512                            TIMER REG
```

Figure 299 Example of Global Numeric Cross Reference List

12.9.3.3 Global Xref - Alphabetic Order

This listing contains the same information as the global cross reference in numeric order, but it is listed in Alphabetic order of the primary symbols. References that have no primary symbol are listed first. Any reference that is configured but not used is not listed. Where more than two assignments are coded for a given Symbol that data and logic element is printed on the next successive line.



```
                                Global Alphabetic Cross Reference List


   REF:    SYMBOL:/DESCRIPTION:   LOCATION:
   000100   COIL100               NC:03S001#0001 NC:03S001#0001 CO:03S001#000
                                  NO:03T001#0001
   000101   COIL101               CO:03S001#0001
   000102   COIL102               NC:03S002#0001 NC:03S002#0001 CO:03S002#000
                                  NO:03T002#0001
   000103   COIL103               CO:03S002#0001
   000104   COIL104               NC:03S003#0001 NC:03S003#0001 CO:03S003#000
   000105   COIL105               CO:03S003#0001
   400012   CYC_HOURS             MC:02P000#0001
             hours
   400005   HOLDREG               01P000#0002    01P000#0002    01P000#0002
             HOLDS THE CONTENTS
             OF THE MULTIPLY
   000007   HOUR_1                MC:02P000#0001
   000008   HOUR_1_24             MC:02P000#0001
_  400013   HOURS24               MC:02P000#0001

   400013   HOURS24               MC:02P000#0001
   000004   HOURS_24              NC:01P000#0001 CO:01P000#0001
             ELAPSED TWENTY FOUR
             HOURS ELAPSED
   400004   HOURS_24OF            01P000#0001    01P000#0002
             CYCLES ACCUMULATES
             NUMBER OF HOURS
   400006   MILITARY              01P000#0002    01P000#0002    01P000#0002
             TIME THIS REGISTER
             SHOWS MILITARY TIME
   000006   MINUTE_1              MC:02P000#0001
   400011   MINUTE_CYC            MC:02P000#0001
   400003   MINUTES               01P000#0001    01P000#0002
             CYCLES MINUTES
   000003   ONE_HOUR              NC:01P000#0001 CO:01P000#0001
             ELAPSED ONE HOUR
             ELAPSED
   000002   ONE_MINUTE            NC:01P000#0001 CO:01P000#0001
             ELAPSED ONE MINUTE
             ELAPSED
   000001   ONE_SECOND            NC:01P000#0001 CO:01P000#0001
_            ELAPSED ONE SECOND

             ELAPSED
   400020   REG20                 03S001#0001    03S001#0001
   400021   REG21                 03S001#0001
   400030   REG30                 03S002#0001    03S002#0001
   400031   REG31                 03S002#0001
   400040   REG40                 03S003#0001    03S003#0001
   400041   REG41                 03S003#0001
   000005   SECOND_1              MC:02P000#0001
   400002   SECONDS               01P000#0001
             CYCLES SECONDS
   400010   SECS10THS             MC:02P000#0001
             seconds
```

Figure 300 Example of Global Alphabetic Cross Reference Listing

12.9.4 Local Cross-reference

When the cursor is on this field, you may elect to have each ladder logic network includes a list of all the coils programmed in the network, and, for each such coil, a list of all the positions in the logic where it is referenced and their state (enabled/disabled, on/off). Your choices are Yes and No i.e., Y or N.

☞ **Note** Local Cross-references are preferred to be printed to the right of each coil in column 12. If there is no room on the page to the right of the rungs, the local xrefs are placed at the bottom of the network.

12.9.5 Network Comments

During a development phase you may elect to review your printed logic without the additional time required to list the comments. This option allows that selection, then toward the end of development or during the commentary phase you can turn the Network Comments print back “on”.

12.9.6 Printing Bit Descriptors

Bit descriptors are displayed in the ladder EDITOR and LISTER in function blocks that reference single bits. Currently, these functions are NOBT NCBT NBIT SBIT and RBIT. For example:

In the ladder editor screen, the symbol and descriptor show in the prompt line as the cursor is placed over a bit value.

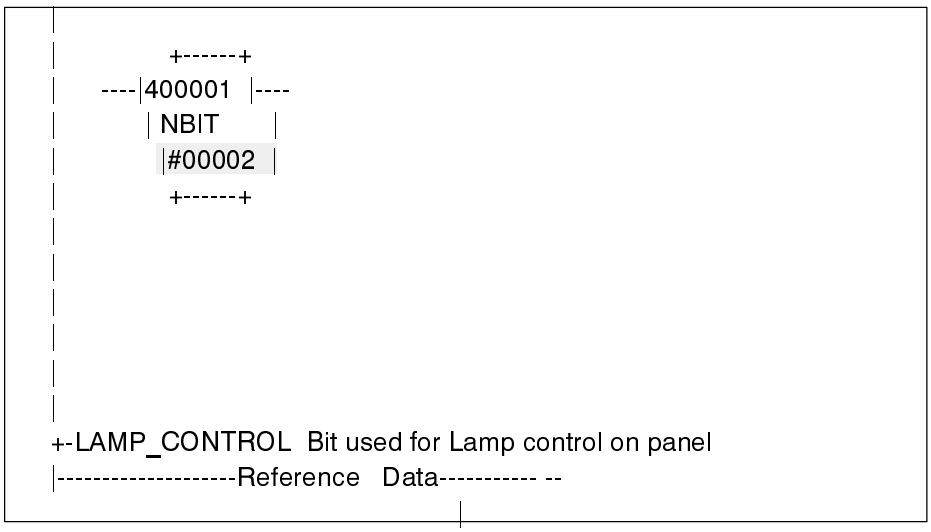


Figure 301 Bit Display

12.9.6.1 Wide Screen Operation

In wide screen mode, the symbols appear as always below the node reference, see the screen below:

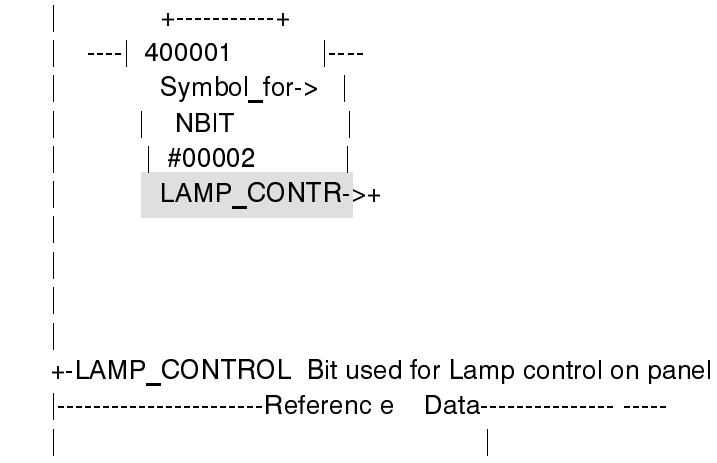


Figure 302 Bit Element With Symbol

In the Ladder Listing printout:

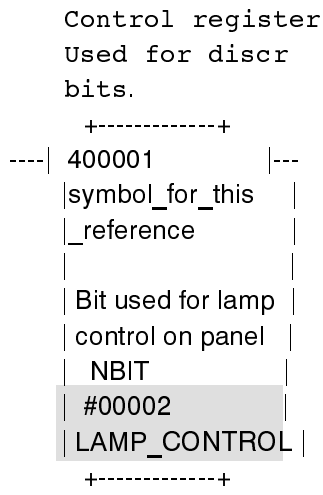


Figure 303 Lister Output Bit Operation

☞ **Note:** This release of modsoft does not include cross reference to the bit level in the cross reference lists. The 4x register itself is already included in the xref lists.

If a bit number is not entered, i.e. you enter only the 400003, with or without the '/', the function is exactly as it currently is, except that the ref field is wider.

12.10 Miscellaneous

The following Misc. options are available:

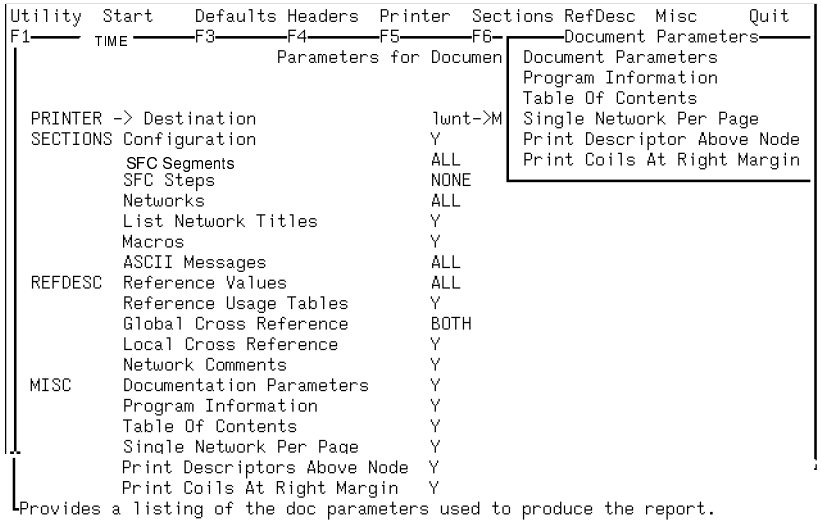


Figure 304 Misc Selected

12.10.1 Document Parameters

This Option can provide you a record of the documentation parameters that are reflected in the present listing.

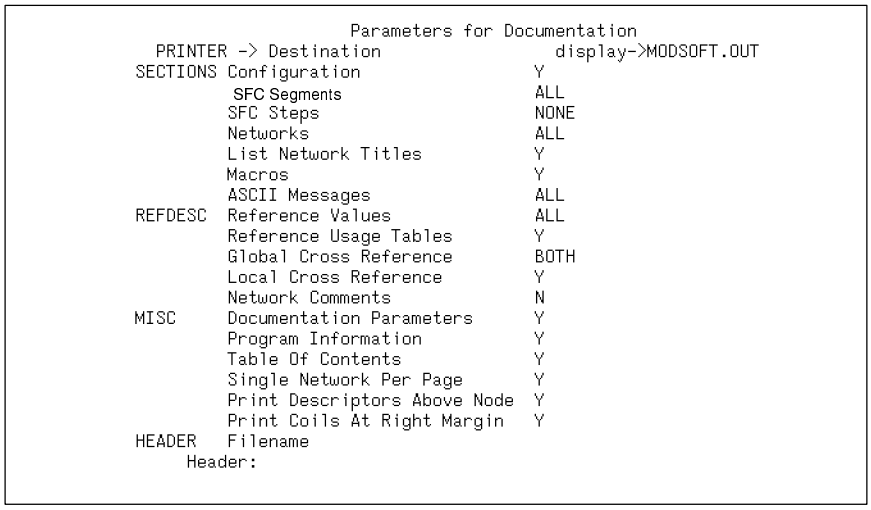


Figure 305 Selections For this Listing

12.10.2 Program Information

If this feature is selected, a summary of the program environment is printed and includes:

Program file names and paths

Revision Identification and reference ranges

PLC specifics including Modsoft Version ID and Segment usage

Communications protocol settings

12.10.3 Support Documentation

The Print function provides additional information in the form of a summary of the *system environment* under which the listing is created, a *TableOfContents* so you can quickly find a specific topic and a list of the *options* you choose for this specific listing. Each of these are illustrated below for your reference.

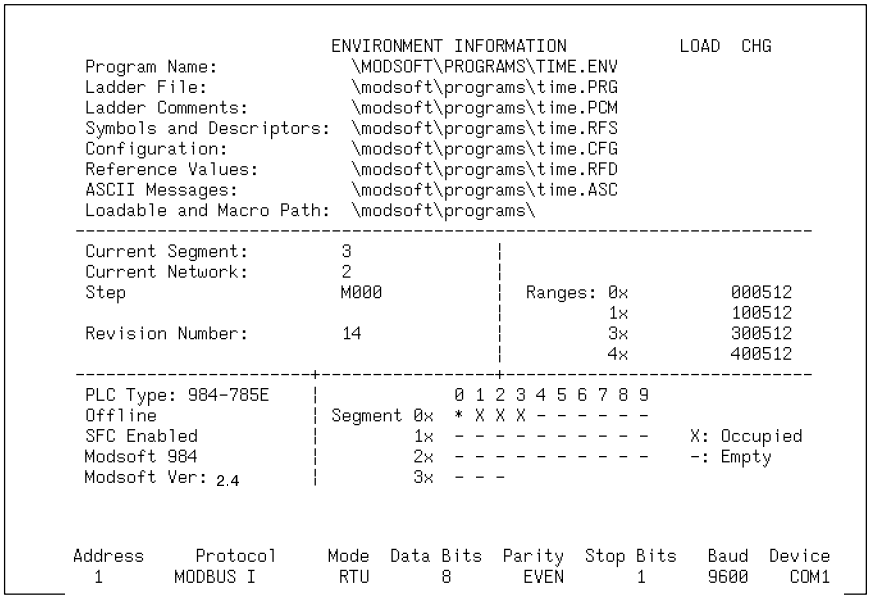


Figure 306 Time Example Environment Information

12.10.4 Table Of Contents

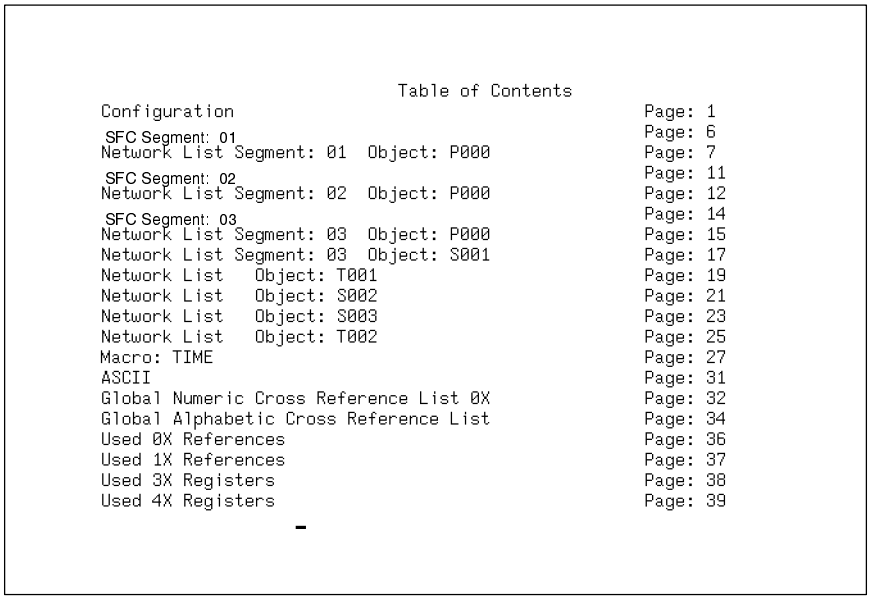


Figure 307 A Table of Contents

12.10.5 Single Network Per Page

If you have a number of networks that do not use all diagram rungs, you may select N (No) for this option which has the effect of using less paper by printing more than one network per page where the space exists to do so i.e., a 4 rung and a 3 rung network can fit on one page. If N is selected, there is a user reference feature “new_page_after_net_comment = 1”, to start the new page after the Network comment if desired.

12.10.6 Descriptor Above node

The affirmative for this feature will cause reference Comments to be printed above the node reference and subsequently symbols below the reference. Selecting the negative puts the symbols above and the comments below. The User Preference File may contain line limits for printing comments and symbols. If a line limit of zero is entered for these () there will be no lines printed for that feature. For example, an entry of ” =0” in the user preference file means no comments will be printed.

12.10.7 Coils At Right Margin

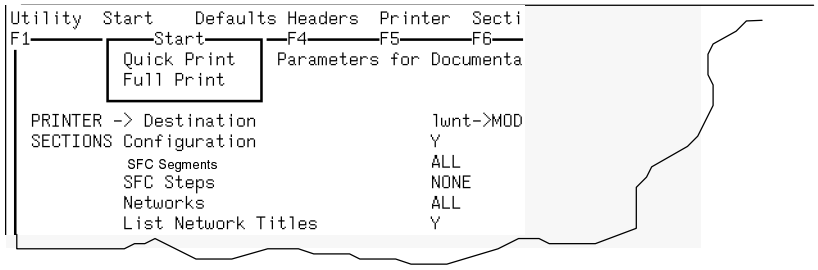
Selecting the affirmative for this feature forces all coils in the network to align in the right column of the network listing, regardless of the column that they are place in the network screen editor. A dashed line connects the last real node with the coil when it is forced to the right of the page.

12.11 Execution

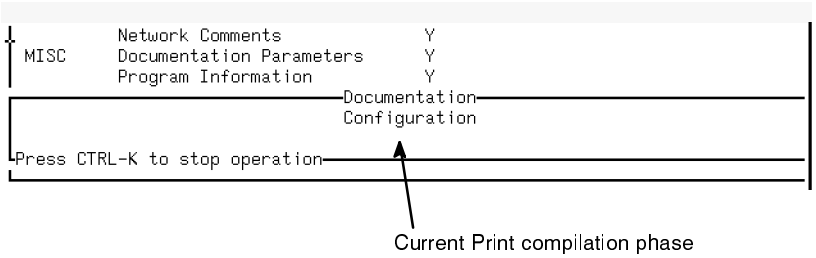
12.11.1 Start Print

When all selections have been made, you start the documentation listing by selecting the menu function "`Start`". is an example of the Start Parameter selection display. You are informed of the various phases that Print performs by posting Print Process screens on the display.

☞ **Note** If you start a Print listing, then want to terminate the process, you can use the <**Ctrl + K** > keys.



Print Start Parameter Select Screen.



Print Compilation Window

12.11.2 Quick Print

Selecting this parameter starts the print compilation for specific networks, therefor you must have ***previously*** (before selecting Quick Print !) set the Network Number in the "`Networks`" variable field and selected the Segment if other than ALL is desired. Although only the se-

lected networks are printed, using the format 1,2,3, ...etc., Lmodsoft compiles all of them to resolve cross-references and pick up proper symbol designations related to the network of interest. this developer aid is much ***faster*** than producing a full default style documentation package.

Your original Print settings are returned to the state they were in after your Quick Print is completed.

12.11.2.1 Hard Copy Output

There are two ways to obtain your hard copy. The listing can be sent to a file and printed after exit to DOS from Modsoft, or it can be sent directly to a printing device that is accessible from your computer. If sending the listing directly to a printer, you will probably be using a DOS device such as LPT1, LPT2, LPT3, COM1, COM2... etc, (see your DOS manual). These names can be entered interchangeably with the file name, only if the printing device has been set up in the UPF editor to have a destination “Device or File via DOS”. If the UPF setup of the printer has selected to be “parallel or serial via BIOS” or one of the NETWORK connections, there is no user choice as to their destinations, listings go to the setup device only. To change destinations in this case, you must use the UPF editor.

Appendix A
Quick Keys

- This list of keys and the topical functions in which they are used can help speed your programming efforts.

A.1 Table of Shorthand Commands

A.1.1 Sequential Function Chart

Alt S Insertion of a step at the actual cursor position
Alt T Insertion of a transition at the actual cursor position
Alt B Insertion of a reference at the actual cursor position
Alt A Single line horizontal link
Alt C Network Comment
Alt P Double line horizontal link
Alt V Vertical link
Alt G Chg Seg
Alt I Initial step
Alt F3 Copy
Alt F4 Delete
Alt F5 Paste
Alt F6 Offset
Alt F7 Search in Program
* Ladder Diagram Editor

A.1.2 Ladder Diagram Editor

" or ' Normally open contact
(or [Coil
/ or \ Normally closed contacts
= (underscore) Horizontal short
| Vertical short
Alt N Negative transition contact
Alt P Positive transition contact
Alt F Inserting a function block
(or just type in the DX name)
Alt F3 Copy
Alt F4 Delete
Alt F5 Paste
Alt F6 Offset
Alt F7 Search
Alt F8 Search Next
Alt F9 Search Previous
Alt C Network Comment Editor
Alt D Delete Network

Alt G	Goto Network
Alt I	Insert Before Network
Alt A	Insert After Network
Alt F2	Reference Data Editor
*	Reference Zoom
Alt W	Screen Size
Alt T	Trace
Alt B	Retrace
Alt Z	DX Zoom
Alt L/M	Memory retentive (Latched) Coil
Alt F1	Symbol Table
Alt O	State Flow
Ctrl F1	Program Information
Ctrl F7	Dos
Ctrl F8	Configuration
Ctrl F4	Save Changes
Ctrl PgUp/PgDn	Toggle Between Segments

A.1.3 Segment Status Display

*	Ladder Diagram
Alt C	Network Comment
Alt T	Segment Comment
Alt L	Segment List
Alt F7	Search in Program
Ctrl F5	Start
Ctrl F6	Stop
Ctrl F1	Program Information
Alt F2	Reference Data
Alt F1	Symbol Table
Ctrl F7	DOS

A.1.4 ASCII Editor

Ctrl F1	Program Information
Ctrl F7	Dos
Alt T	Text
Alt C	Control Code (octal)
Alt V	Variable (ASCII, Binary, Decimal, Hexadecimal, Integer)
Alt P	Space
"/"	Carriage Return
Alt F	Flush

Alt A Repeat
Alt F1 Symbol Table
Alt F2 Reference Data
Alt F3 Copy
Alt F4 Delete
Alt F5 Paste
Alt L Load Message
Alt S Save Message

A.1.5 Network Editor

Alt F3 Copy
Alt F4 Delete
Alt F5 Paste
Alt F6 Offset
Alt C Network Comment
Alt G GoTo Comment
Alt I Insert Before Network
Alt A Insert After Network
* Zoom to Ladder Editor

A.1.6 Macro Editor

Alt F2 Reference Data Editor
Alt F3 Copy
Alt F4 Delete
Alt F5 Paste
Alt F6 Offset
Alt F7 Search
Alt F8 Search Next
Alt W Screen Size
Alt L Load Macro
Alt S Save Macro
* Macro Ladder Diagram
Alt C Macro Network Comment
Alt G Goto Network
Alt I Insert Network
Alt A Add Network
Plus the Ladder Editor Shorthand Keys and * for Reference Zoom

A.1.7 Reference Data Editor

Alt A ASCII Format
Alt B Binary Format
Alt D Decimal Format
Alt C Clear Screen
Alt E Floating Point
Alt K Signed Long
Alt X Hexadecimal Format
Alt M Matrix Format
Alt N Range
Alt W Change Window Size
Alt S Save Template
Alt U Long Decimal
Alt I Signed Integer
Alt L Load Template
Alt F1 Symbol Table
Ctrl F7 Dos
Ctrl F1 Program Information
+ Increment Reference by 1
- Decrement " " "
= Enter the Same Reference Again

A.1.8 Symbol Table

Alt I Insert
Alt F1 Symbol Table
Alt F2 Reference Data
Alt F3 Copy Comment
Alt F4 Delete
Alt F5 Paste Comment
Alt F6 Offset
Alt F7 Search
Ctrl F7 Dos
Ctrl F1 Program Informationc

A.1.9 Keys of General Importance

Tab	Toggle between menu bar and editor screen
Esc	Stop entry or back out one menu
Enter	Select an entry or Finish an input
F1 - F9	Select the respective entry from the Menu Bar

F10	Leave current editor for previous editor
Shift F1 - F9	Help on the respective entry from the Menu Bar
Alt H	Help on current entry on menu bar
?	Popup selection list on entries is available
Space	In Ladder or Reference Edit , use same value
Ctrl-F	Toggle use of Function Keys
Alt C	Clear Screen
Ctrl-K	Abort Lister or Modbus Communication
Ctrl-Home	First Network in a step
Ctrl-End	Last Network in a step or segment
Home	First Network in Netlist
End	Last Network in Netlist

Appendix B
Op and Comm Error Codes

- Op Codes define computer instructions. The following defines the opcodes assigned to a set of Loadable Instructions.
- Comm error codes are now largely reported as text based statements. However; some errors include a numeric designator. This section represents those numeric designators that may occur in a communications sequence.

B.1 Op Code Assignment

The following table lists A/B/X and -8 Op Code assignments as of the date of publication of this document.

DX Opcodes

984 A/B/X Loadables	Opcode (Hex)	Basic	Enhanced 984-8 Loadable
PID	5E		PID2
XMWT 984-B (only) Xmem Write	7E	Generic I	Generic I
XMRD 984-B (only) XMEM Read	9E	Generic II	DRUM
MATH/EARS	BE	LAB	LAB
DMTH	DE	JSR	JSR
DRUM	FE	RET	RET
MBUS/MAP3	1F		MBUS/EUCA
PEER	3F		PEER/HLTH/MAP3
CALL/HLTH	5F		FNXX/CALL/ICMP/EARS
ICMP	7F		EMTH ICMP
BLKT	9F		BLKT
CKSM	BF		CKSM (or MSTR)
TBLK/EUCA	DF		TBLK
HSBY	FF		HSBY/DRUM

☞ **Note** The 984-78X allows additional opcodes; All opcodes above 1B hex; e.g., 20, 21, 22 - 2F; 30, 31, 32-3B; 40-4F; 50-5B; 60-6F; 70-7B; 80-8F; etc. to F0-FB.

B.2 Comm Error Code

000 Online Error 001 Modcom II (c) database corruption detected 003
Modcom II (c) does not support this function 004 A response was not obtained 005
User initiated communications abort 010 Invalid mode - Must be 0 (ASCII) for VMS 011
Invalid mode - Must be 0 (ASCII) or 1 (RTU) 012 Bad port - The requested number is out of range 030 Invalid slave ID (Slave ID must be 0-247) 031 Invalid Channel (This is not a valid port address) 032 Invalid type (Must be a valid PLC designation) 033
Broadcast illegal 034 Attempt to read packed 484 coils across a 257 boundary 040
Illegal reference start or size specification 041 Illegal subfunction code passed to Modcom II (c) 042 Unable to read 984 Dx table 101 Function code not supported by PLC 102
PLC rejected start address as illegal 103 PLC rejected transferred data as illegal 104
PLC failed to respond, or abortive error 105 PLC returned an ACK exception (request for POLL) 106 PLC returned a BUSY (unable to service request) 107
PLC returned a NAK (Cannot understand request) 108 PLC returned a memory parity error 109 PLC returned undefined exception status 110 Host received an invalid response buffer 111 No start of response 112 Invalid ASCII character in response 114
Unexpected address/function code received 116 Unexpected response length returned 120
Bad checksum on response 129 No start of text on program response 144
Invalid NAK response 145 Bad program command checksum 146
Length is truncated 160 984-Invalid command 161 984-Invalid address/address range 162 984-Memory protect on 163 984-Memory full 164 984-Invalid node type 165
984-Coil is used 166 984-Invalid reference number range 167 984-Invalid parameter 168
984-Controller running 169 984-Not logged in or a user is logged in on another port 170
984-Coil not disabled 171984-Search failed 172 984-Illegal configuratoin 173
984-Illegal message 174 984-Illegal sequence of insert 175 984-Constant sweep time base missing 176 984-Coil disable not allowed 177 984-cannot modify PROM memory 178 984-Coprocessor failure 190 984-No ASCII message with this number 191
984-ASCII message number previously used 194 Map driver is not installed or is not functional 195 Unable to allocate enough space 196 Executive failure 197
Pipe Failure 198 Write Failure 199 Timeout without a PLC response - Review service panel port selections 499 Invalid parameters to command 501 Link Status-Illegal function or orphan poll 504 Link Status-PLC failed to respond or abortive error 506
Link Status-PLC BUSY. Function can not be performed at this time 511
Link Status-Message size greater than PLC. 512 Link Status-New message started before previous one completed. 513 Link Status-Packet out of sequence. Command aborted. 514
Link Status-Major sequencing error detected. Reset link required 515
Link Status-Response is to large for PLC. Command aborted. 516
Link Status-PLC returned a reserved response. Unrecognized PLC response. 600 Map 3 application source SSAP not in range 601
Registering with common environment failed. 602 Blocked Read error. 603

Unable to allocate EPB 604 Common environment Read error. 605
Blocked Reads disabled. 606 Common environment returned a Write error. 607
No common environment resources for write. 608 Common environment response error 609
Error allocating buffer 610 Error de-registering 611 MAP 3 not opened 612
Link layer error 613 Unexpected response type 614 Error from remote device 615
Reject PDU received 616 Unexpected Message 617 Error during deactivate 618
File config.adr could not be opened 619 Error returned from packing data 620
Error returned from unpacking data 621 This task has a SAP already in use 622
Invalid service type requested in open. 623 Address not configured in config.adr 624
Timeout writing to link layer 625 Timeout writing from link layer 626
Error de-allocating EPB 627 Error reading immediate acknowledgement. 628
Controller is in dim awareness- check and reload configuration 630
Invalid function for this mode. 700 Request could not be performed 750
DecNet-Send error. Unable to communicate with device driver 751
DecNet-Receive error. Unable to communicate with device driver 752
DecNet-Can't open Modbus Plus path 753 DecNet-Modbus Plus send error. 754
DecNet-Modbus Plus receive error. 755 DecNet-Co-Pro backplane read/write error 756
DecNet-Unknown Modbus exception condition 757 DecNet-Illegal Modbus function or message format 800 Modbus Plus-Unable to communicate with device driver 801
Modbus Plus-Bad send or status buffer size 803 Modbus Plus-Invalid Net BIOS command 805 Modbus Plus-Command timeout has expired 806 Modbus Plus-Receive buffer not big enough 808 Modbus Plus-Bad session number 809 Modbus Plus-LAN card does not have enough memory 810 Modbus Plus-'Session closed 811
Modbus Plus-Cancel 813 Modbus Plus-Name already exists in this PLC 814
Local name table is full 815 Modbus Plus-Cannot delete name used in session 817
Modbus Plus-Local session table is full 818 Modbus Plus-Remote PLC not listening for call 819 Modbus Plus-Bad name/number 820 Modbus Plus-No answer 821
Modbus Plus-No such name in local name table 822 Modbus Plus-Name is in use elsewhere in network 823 Modbus Plus-Name incorrectly deleted 824
Modbus Plus-Session aborted abnormally 825 Modbus Plus-Two or more identical names in use 826 Modbus Plus-Bad NetBIOS commands queued 833
Modbus Plus-Network card is busy 834 Modbus Plus-Too many NetBIOS commands queued 835 Modbus Plus-Bad card number 836 Modbus Plus-Command finished while cancelling 938 Modbus Plus-Command can't be canceled 899 Modbus Plus-Still processing command

Appendix C
Setting Up a Client Node for Modsoft

- Setup Examples

C.1 Setup Examples

This Appendix provides examples of how you can configure an IBM PC as a Decnet client node for using Modicon Modsoft panel software with the gateway.

Caution The setup parameters in this Appendix are required for Modsoft compatibility with your Ethernet gateway. In addition to using these parameters for Modsoft, you must determine their compatibility with the overall functioning of your Decnet.

Before placing any node online with these parameters, consult your network's System Administrator. Determine the effect of adding the node with these parameters, or the effect of changing the parameters of an existing node.

These examples also show how you can check the client's communication with the gateway. You should complete the setup of your client node with these examples, before you install Modsoft on the node.

C.1.1 Setup Requirements

The client node must have the following configuration to support Modsoft access to the gateway:

- Digital Equipment Corporation's PATHWORKS, version 4.0
- PC hardware: 20M hard disk required; 80386 CPU or higher recommended
- DOS version 5.0 or later.

C.1.2 Disks and Paths in These Examples

The examples in this Appendix include DOS disk and directory paths. If these example paths are different from the paths you are using, modify the examples to agree with your PC.

C.1.3 1. Configuring CONFIG.SYS

Modify the client node's CONFIG.SYS file to include the following parameters:

```
device=C:\DOS\SETVER.EXE
device=C:\DOS\HIMEM.SYS
DOS=HIGH
buffers=25
files=25
shell=\COMMAND.COM /P /E:1000
device=C:\Decnet\PROTMAN.SYS /I:C:\Decnet
device=C:\Decnet\MACWD.DOS
break=ON
lastdrive=Z
device=C:\DOS\ANSI.SYS
```

C.1.4 2. Defining the Client Node

Define the client node's Decnet node address and name. If the node is a new one, without a current address and name, you must assign one. You can obtain this information from your Network Administrator. If you have access to a node with the Decnet Network Control Program (NCP), you can list the network's existing nodes:

```
C:\Decnet\NCP LIST EXECUTOR
```

Write down the address and name of the client node:

Example:	3.13	(address)
	Client13	(name)

C.1.5 3. Saving the Client Node's Decnet Files

Three files in your client node are needed for Modsoft. Their filenames are: Decnet.INI, PROTOCOL.INI, and DECPARM.DAT.

The contents of the two .INI files must be verified to make sure they use the correct parameters, and they must be edited as required. It is recommended that you use the DOS COPY command to make a backup copy of each file before changing any parameters. For example:

```
COPY Decnet.INI Decnet.OLD
COPY PROTOCOL.INI PROTOCOL.OLD
```

The DECPARM.DAT file will be created as a new file when you activate the client node (later in this Appendix). If a DECPARM.DAT file already exists in the client node, you must rename it to allow a new file to be created. For example:

```
RENAME DECPARM.DAT DECPARM.OLD
```

C.1.6 4. Modifying Decnet.INI and PROTOCOL.INI

PROTOCOL.INI and Decnet.INI are text files that you can modify using a local text editor. Your gateway disk provides an example of both files. Note that the terms of your gateway software license agreement do not allow you to use these files directly, or to use a copy of them, on any node other than your gateway node. Therefore the following steps only describe how you can use them as examples.

Use the DOS TYPE command to view the files on the gateway disk, or use the PRINT command to make a hard copy of them. These files show the parameters that are required in your client node's files for using Modsoft. Compare these example files with the files in your client node. You may have to modify some of the parameters in your client files.

Decnet.INI Using a text editor, modify the client's Decnet.INI file to make it identical to the example on your gateway disk.

PROTOCOL.INI Using a local text editor, first modify the client's PROTOCOL.INI file to use the same parameters as the gateway's example file. Then modify it to agree with the hardware setup of your client's Decnet controller board.

If you modify the file for compatibility with your Decnet, take care not to change any fields that are required by Modsoft. The example below shows which fields you must not change in the file, and which fields must agree with your hardware.

```
[PROTOCOL MANAGER]
DRIVERNAME = PROTMAN$

[IPX$MAC]
DRIVER = IPX$MAC]
BINDINGS = MACWD.DOS

[DATALINK]
DRIVERNAME = DLL$MAC
LG_BUFFERS = 64                    (Do NOT change)
SM_BUFFERS = 32                    (Do NOT change)
OUTSTANDING = 32                   (Do NOT change)
HEURISTICS = 6                     (Do NOT change)
BINDINGS = MACWD.DOS
NI_IRQ = 5                         (Agree with Hardware)

[MACWD.DOS]
DRIVERNAME = MACWD$
IRQ = 5                            (Agree with Hardware)
RAMADDRESS = 0XD000                (Agree with Hardware)
IOBASE = 0X280                     (Agree with Hardware)
RECEIVEBUFSIZE = 2048              (Do NOT change)
```

C.1.7 5. Creating a Batch File for the Client Node

Create a batch file called EMBP.BAT to contain the commands for activating the client node. Place the file in the root directory C:\. The file's contents are shown below. Modify the paths as required for your PC.

```
echo off
cd \decnet
dllndis
netbind
sch /h
dnnethat
if not exist \decnet\decparm.dat goto firsttime
goto end

:firsttime
echo The Decnet is not configured yet - please wait.
cd \decnet
ncp < \decnet\decnet.ini
echo .
echo Define your Decnet address and name now.
echo .
echo This is only an example - use your actual parameters:
echo .
echo     ncp
echo       define exec address 3.13
echo       define exec name client13
echo       exit
echo .
echo Please reboot the client computer for the
echo Decnet initialization to take place.

:end
echo To use Modsoft type
echo \MODSOFT\MODSOFT
```

C.1.8 6. Activating the Client Node

After creating the batch file, reboot your client (with Ctrl-Alt-Del). Then run the batch file. When prompted for your Decnet address and name, enter the client's address and name that you defined earlier in this Appendix.

After running the batch file, reboot the client again to activate it.

C.1.9 7. Checking the Client's Configuration

Check the client configuration using the following Decnet Network Control Program (NCP) command:

```
CD\Decnet
NCP LIST EXECUTOR CHARACTERISTICS
```

If `MAXIMUM LINKS=32` your client is configured correctly.

C.1.10 8. Checking the Client's Communication With the Gateway

The following Decnet Network Control Program (NCP) tests can be used to check communication between the client node and the gateway.

1. Install the gateway and start it running as described previously in this guide. Then exit the gateway application by typing `q`.

2. At the gateway, use these NCP commands to check the gateway internally:

```
CD\Decnet
NCP ZERO EXECUTOR             (Zeros all executor counters)
NCP SHOW EXECUTOR COUNTERS    (Make sure there are no errors)
NCP SHOW LINE COUNTERS        (Make sure there are no errors)
NCP SHOW CIRCUIT COUNTERS     (Make sure there are no errors)
```

3. At the client node, run the same set of NCP commands as listed in Step 2 to check the client internally.

4. Start the NCP Mirror routine on the gateway and let it run:

```
NCP MIRROR
```

5. At the client node, use these NCP commands to check communication to the gateway:

```
NCP LOOP NODE x.yy COUNT 10000 LENGTH 128
```

where: `x.yy` = the gateway's Decnet address.

```
NCP SHOW EXECUTOR COUNTERS    (Make sure there are no errors)
NCP SHOW LINE COUNTERS        (Make sure there are no errors)
NCP SHOW CIRCUIT COUNTERS     (Make sure there are no errors)
```

6. Repeat Step 5 for each client node that will be running Modsoft. When all tests have been completed without errors, you can install Modsoft on the clients. Follow the procedures in your Modsoft guide for installing Modsoft.

AppendixD
User Defined Operation

Expert The User Preference File provides a method for setting Modsoft defaults that occur each time modsoft is run. When you start Modsoft, if a file named `Modsoft.UPF` is found in the current working directory, it is parsed and the information contained is used to initialize Modsoft. The .UPF file takes precedence over the command line switches.

The *Format* of each line is LABEL=VALUE where the value must be enclosed in double quotes if it contains any spaces. For example font_weight="extra thin". Each label should begin on a separate line. The order of lines in the file does not matter; however, the file is processed in order, and the last command takes precedence if there is a conflict.

This Appendix instructs you on using the UPF Editor to create the .UPF file and how to make changes.

D.1 User Preference File

Modsoft allows you to customize your functionality by setting variables in a file called the User Preference File. You can make most of the standard entries using the user interface screens provided when executing the .UPF program in the Modsoft director, some others require an ASCII editor applied to the named Modsoft file with a .UPF extension. Any setting that is available as a /switch can be included. Old DOS "Set" commands can also be included. There are also additional functionality settings available including:

Saving a file to an automatic backup.
The ability to turn off "`Are you Sure Prompts`"
Setting defaults between Powerflow and State flow.
Establishing Printing Preferences

The following table gives you key names and equivalents for the Preference statements.

User Preference File Statement Summary

Label	Meaning	Integers	Default
names_directory	Default directory Where .Env file is found		\modsoft\programs
files_directory	Default directory Where files are found		\modsoft\programs
rt_directory	Runtime directory path		\modsoft\runtime
tmp_directory	Temporary directory path		\modsoft\tmp
dos_path	Storage for DOS swap space		\modsoft\tmp
user_level	Set user level 0-8		8
screen	Color vs Monochrome	0=automatic 1=mono, bold 2=mono/w reverse video	0
powerflow_style	Bold vs reverse video	1=reverse	0 = Bold
stateflow	Show Ladder state flow	1=state	0 = powerflow
allow_tab	Allows use of tab to menu bar	1=enable	1
quick_Keys	Enable use of quick keys	1=enable	1
reference_style	How many digits	5=5	Automatic
		6=6	

Label	Meaning	Integers	Default
auto_backup	Write current file version to backup directory at each save	1=backup	0
warn_at_main	Warn if return to main w/o save	1=warn	1
recover	Equivalent of /r	1=recover	0
recover_time	Equivalent of /rn	time	0
journal	Equivalent of /j	1=save to .jou	0
using_sfc	Opposite of /p	1=use sfc	0
used_refs	Equivalent of /u	1=do not save used references	1
modbus_plus_interrupt	Equivalent of /v	Interrupt vector	5c
ask_for_confirmation	Confirm keystroke	1=turned on	1
ladder_auto_advance	Advance cursor *after*	1=auto	1
use_m_for_latch	show latched coil as -(M)-	1=-(M)-	0
communications_timeout	range 1 - 99	number	20 (sec)
disallow_tracking_help	Save time on display window	1=on	0
immediate_command	Equivalent of /o Online — Immediately select Online Program Offline — Takes you directly to offline ladder editor Combined — Takes you directly into the PLC Verify — Verifies the PLC and Program Status — Gives the status of the selected PLC File_to_PLC — Transfers the program file to the PLC PLC_to_file — Transfers the contents of the PLC to a file. PRINT — Execute the Print function		Online

Label	Meaning	Integers	Default
iec_standard	If 0 do not follow iec std symbols		0
engr_flag	Set Engineering Flags		0
trace_modbus	Save communications with PLC in Modsoft.Bus file		0
segment_list_mode	Displays segment status screen with segment list information		0
user_level	Set the user access level for this execution of Modsoft. Higher levels include the functionality of all lower levels. 0 Monitor only 1 RDE Online changes 2 Allow PLC to Program transfer 3 Allow PLC Start, Stop 4 Allow Program to PLC transfer 5 Save offline editors 6 Allow full Programming 7 Same as 6 8 Allow Executive downloads		8
reference_style	If 5, use only 5 digit references If 6, use only 6 digit references		auto
track_memory	Save all internal memory allocations to a file		0
ladder_full_screen	Display ladder in full screen		0
word_wrap	Allow word wrap in printout		on
font_weight	Weight of printed copy. One of: Ultra Thin, Extra Thin, Thin, Extra Light, Light, Demi Light, Semi Light, Medium, Semi Bold, Demi Bold, Bold, Extra Bold, Black, Extra Black, Ultra Black.		medium
font_style	Style of printed copy. One of: Upright, Italic, Slant, Superscript, Subscript, Outline, Shadow, or Shadow and Outline.		Upright

Label	Meaning	Integers	Default
font_typeface	This is the selection of print typefaces: Line Printer, Helvetica, Triumvirate, Sans, Futura, ltc Avant Garde Gothic, Fusion, Triumvirate (historical), Orator, Presentation, Optima, Univers, Galixy, Antique Olive, Frutigher, Roissy, Eurostyle, Microgramma, Letter Gothic, Franklin Gothic,News Gothic, OCR a, OCR b, Courier, Prestge, Picaelite, Typewriter, Times Roman, CG Times, Roman, Garamon, Bookman, Palatino, Centry Scholbook, Baskerville, Bodoni, CG Bodini, Script, Park Avenue, Brush Script, Kaufmann, Ariston, Slogan, Zapf Chancery, Brodway, University Roman, American Unical, Blipo, Old Town, Sencil, Old English, Mariage, Cloister, m PS Symbol, Dingbats.		line printer
mouse_vertical_speed	mouse cursor rate	1-32762	
mouse_horizon-tal_speed	lowest number is most sensitive	1-32762	
disable_mouse	Disable Mouse		1
upload_to_file	On upload, put uploaded code into a file		1
segment_list_mode	Display the segment sta-tus screen with segment list information		
export_tag_mark	Special setup strings for export of references to MMI such as Unicell. A typical mark might be “//” for instance.		Blank
lister_max_cmnt_lns	Maximum number of lines of comments to be included above (or be-low) nodes in Network listings.		All
lister_max_sgmt_lns	Maximum number of lines of Symbol in Net-work listing.		All
lister_max_xref_lines	Maximum number of crossreference to right of coils in Network listing		All

Label	Meaning	Integers	Default
new_page_after_net_comment	Cause a page break after network comment even in multiple nets per page method.		0
iec_standard	International Electrical Code		1
set_p190_coil_mode	Show all coils in column 11.		0
symax_s908-io-mode	Use SyMax S908 I/O		0
disab_ref_chg_hotkey	Disable the reference change hot key ","	1 Disables 0 Enables	1
duplicate_coils	Allow download of duplicate coils, and insert of duplicate coils online.	1 = On	0
set_fast_panel	Use special treatment for power flow display when PC is high speed	1 = Fast Panel	0
disable_mouse	Shut off the Modsoft mouse	1 = Disabled	0
y_n_q_safety	Instructs Modsoft to use safe defaults Y/N	1 = Safe	0
disab_ref_chg_hotkey (not in UPF)	disables Modaoft use of the "," hotkey in the Ladder editor	1 = disable	0

Note The "," hotkey is a programming convenience key. It is intended to provide a way for a programmer to lookup a reference by its symbol and description in the Table Of Symbols via the reference symbol editor; and upon returning the reference is inserted in a ladder logic node.

Caution: In the On-line mode there can be unwanted side effects with the "," hot key. It is recommended that it be left disabled.

D.2 The UPF Editor

A customized Modsoft.UPF file can be created and edited using the `UPF Editor`. This Editor not only provides User Preference Templates for you but also includes the ability to customize the printer used to output your ladder logic listings
(Modsoft .pdb).

☞ **Note** Individual settings may still be evoked as a command line switch.

D.2.1 Editor Interface

The UPF Editor is started from the Modsoft directory. Your distribution disk will automatically place the file UPF.exe in the Modsoft directory. To start the Editor type:

C:\MODSOFT>UPF↵

The Main screen and Menu items are displayed as:

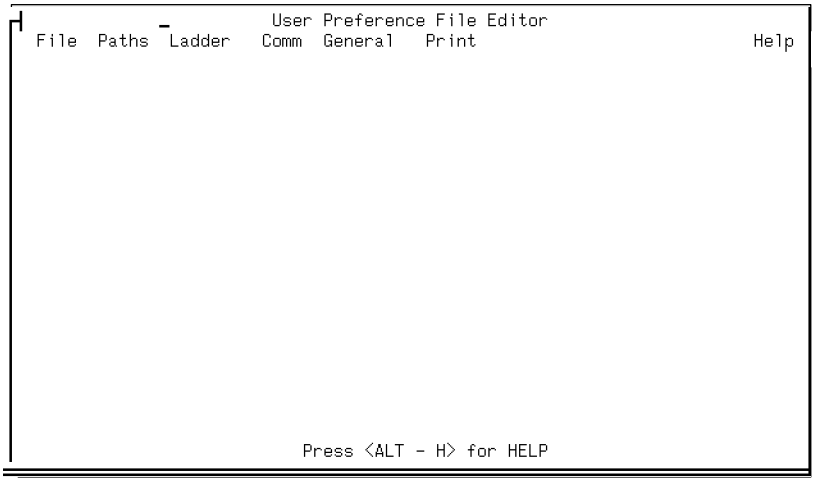


Figure 308 UPF Editor Initial Entry Screen

To begin using the Keyboard press the <**Alt**> key to highlight the *quickkeys* you can utilize in the menu selections . You can now use *arrowkeys* with the **return** key or the **Alt Key + L**etter to move from field to field. If you press **Alt + H** for Help, you will see a summary of the functional keys for this interface. Press the "**x**" key to return to the main menu. This executable can also be used with a mouse.

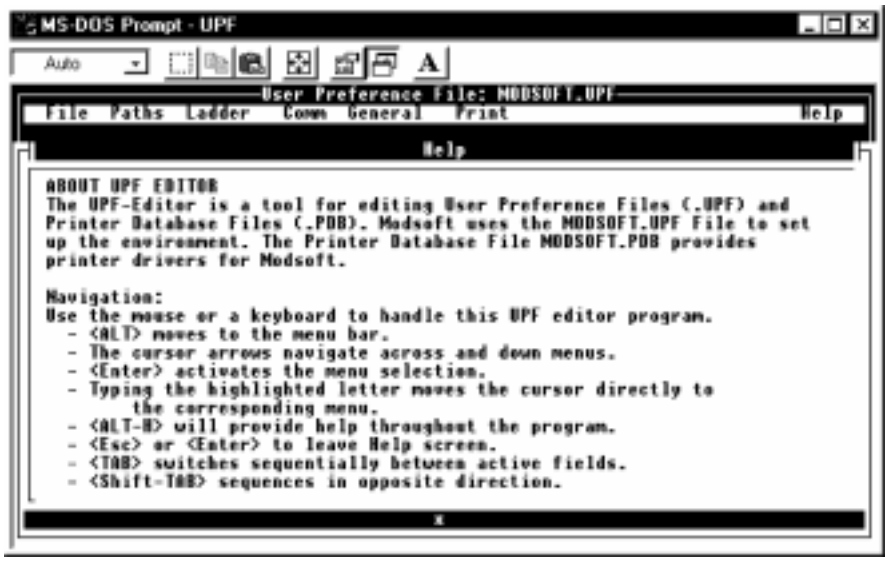


Figure 309 Top Level Help Display

D.2.2 Menu Selections

Primarily, the User Preferences and User Levels are selectable from the UPF Editor Main menu pulldowns (the `Print` entry is documented in the Print Chapter). The Pulldowns and the associated Quick Keys are illustrated in the following paragraphs.

D.2.2.1 File

The `File` pulldown includes the typical file handling functions found in file related systems. Most of the Functions you encounter here have been documented in the Modsoft Offline Function where Opening Closing or renaming a file copy (Save As) can be found. The DOS option is the same as Modsoft i.e. when taken, type **exit** ↵ to return. If you forget how you arrived at the DOS prompt, and try to execute a program you may see the “Not enough Memory” Error.

In addition, to find the version level of the Editor press **b** or put the cursor on the A**b**out field and press the return key. E**x**it returns to the main menu.

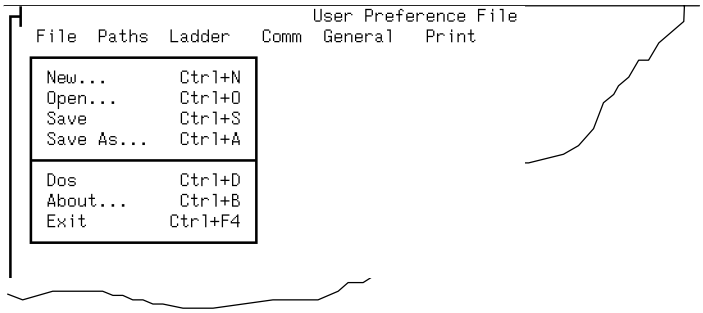


Figure 310 UPF Editor Files Pulldown Select Functions

D.2.2.2 Making a Modsoft.UPF file

Whether you have an existing file to `Open` or need to create a `New` one you will be assisted by a screen dialog.

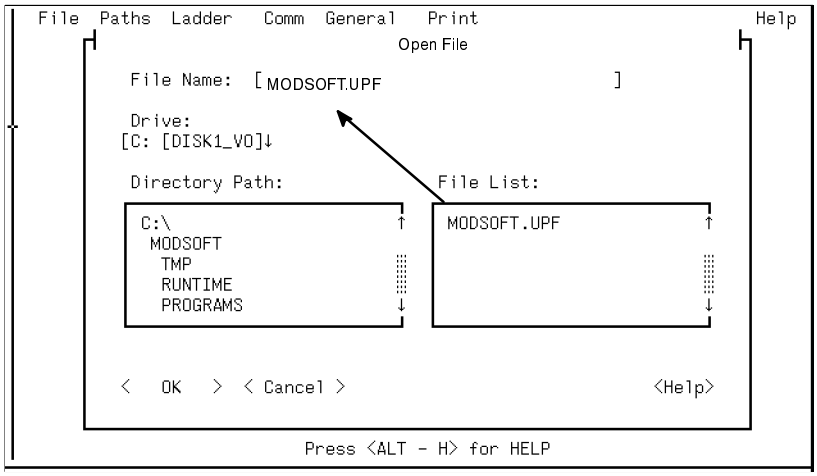
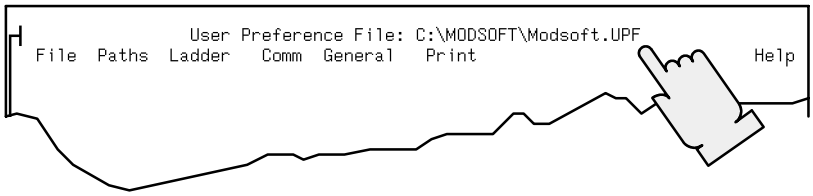


Figure 311 Typical File Dialog

☞ **Note** The file name you are looking for must exist and be a legal name. Either double click on the Listed file or use the <Alt><Tab> <↓ or ↑ >,<Enter> sequence to select it. The selected file name is posted in the File Name field.

In this example the file name **Modsoft** had previously been edited in the filename field and is the default. The .UPF extension is fixed in the field. Note that the lower case characters are converted to upper case.

Once you have entered the filename the file is created in the directory set in the variable directory field and a banner is posted to the top display line that informs you of the current file name and path.



During the UPF development if you enter any of the template variable fields, a change is signaled and you will be advised to ***save changes*** if you try to exit the edit session without saving the file directly in the `File` functional pulldown.

D.2.2.3 Paths

Each of the Modsoft primary directory ***types***, have a 31 character field that you can specify. If you have no specific requirement to change the Modsoft Path\File routing, you can simply elect to continue with the standard defaults which are:

```
Files - Runtime - TMP
```

The distribution disks for Modsoft, locate these directories as a subdirectory to the directory `MODSOFT`.

```
Environment
```

The default On the distribution disk is to the MODSOFT directory. But, in the execution of Modsoft the .ENV files are established in conjunction with the internal file settings which are \Modsoft\Programs unless otherwise specified.

The DOS swap file at default uses The TMP directory

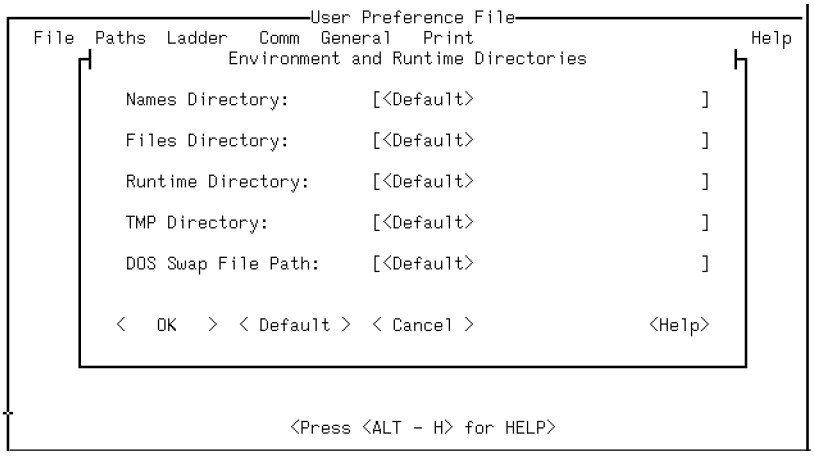


Figure 312 UPF Editor Paths Pulldown Select Functions

D.2.2.4 Ladder

This template produces file characteristics for display and printing of networks. as you enter each variable field, the **arrow keys** allow to toggle to the choice desired. The choice selected is defined with a bullet within the parentheses.

Figure 313 UPF Editor Ladder Pulldown Select Functions

Warning: If you choose reusable Coils, a message is displayed

"reusable coils are potentially dangerous. Use with extreme caution."

Controllers supporting the re-use of coils are the Quantum and the "E" controllers. This option is identical to the /d (duplicate-coil) startup switch, and the modsoft.upf file entry `duplicate_coils = 1`. It is now capable of being selected in the UPF editor screen. **Allowing Duplicate or re-usable coils is a potentially dangerous operation. The user must be aware that they should only be used in places which do not present a danger if energized inadvertently.** Due to this danger, A warning is issued each time you enter the LADDER screen and the .upf `(.) Yes` has been selected. The same warning also occurs when you change the selection from `(.) No`. When Reusable coils are selected, Modsoft allows you to set a starting coil value above which the coils are re-usable. Modsoft continues to monitor and warn you when downloading duplicate coils. Find the Re-usable coils range selection in the Configuration overview screen.

D.2.2.5 Comm

This dialog allows you to set the Modbus Plus interrupt to some hex value other than 5C. In this screen you can also alter the elapsed time value after which to an error should be set. This feature is provided for cases where this interrupt may be assigned to another function, in which case this dialog allows you to avoid the conflict. *This is particularly important and may need to be changed when using windows and have installed a Modbus Plus card.*

Figure 314 UPF Editor Comm Pulldown Select Functions

D.2.2.6 General

The General function pulldown provides four common preference templates that you fill in to implement selected preferences.

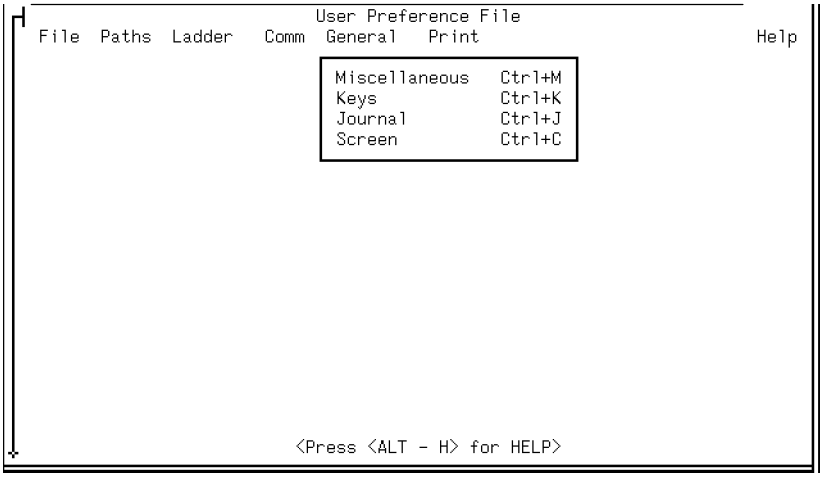


Figure 315 UPF Editor General Pulldown Select Functions

- **Miscellaneous** - This template combines the User Level operating constraint selections illustrated and the Immediate operating characteristics you want demonstrated by Modsoft on start-up. These selections are in the highlighted fields and are posted to the field using the ***up*** or ***down*** arrow key.

Figure 316 Miscellaneous UPF Template

- **Keys** - This template allows your preference for keystroke options, warning and help message display. If using a mouse you can also set the reaction speed in this display.

  There are several places in the modsoft operations which are potentially hazardous if the user selects the default YES or NO answer to a warning message, by simply pressing the ENTER key. It is generally considered ”convenient” to accept the default answer. When safety is at stake, this option requires the user to change the answer to cause the action to be completed. Most frequently the “convenient” answer is Yes, and the “Safety” answer is No. In some cases the opposite is the case. If the “Safe” option is selected here, Modsoft provides, as the default, the answer that will **NOT** complete the action if the user simply presses the ENTER key.

Figure 317 Key and Warning Preference Template

If you have chosen to enable the “,” hot key, you can maximize safety by choosing both Confirm Operation = Yes and Safety Y N defaults = Safe. This setting has the effect of prompting you to confirm the potential reference change and the default is “N” thus forcing you to press “Y” to implement the change.

Modsoft’s use of the mouse is a carry over from older DOS applications, and is not consistent with the most modern use of the mouse as a pointing and selecting device. If you choose you can disable its use in modsoft. This does not disable use of the mouse in the UPF editor or in any other application or Windows, only in modsoft. When the mouse is disabled, none of the other mouse related vertical and horizontal speed numbers have any meaning.

- **Journal** - This Template allows you to record the key strokes you make to traverse a Modsoft menu path. You can play “Recover” the record (in the .jou file) with modsoft reaction exactly as first captured. The time between key and execution sequences is variable

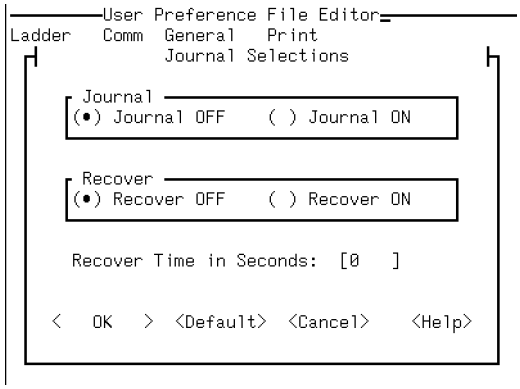


Figure 318 The Journal Dialog Preference Settings

- **Screen** - This template allows you to customize the color of the various display windows in Modsoft.

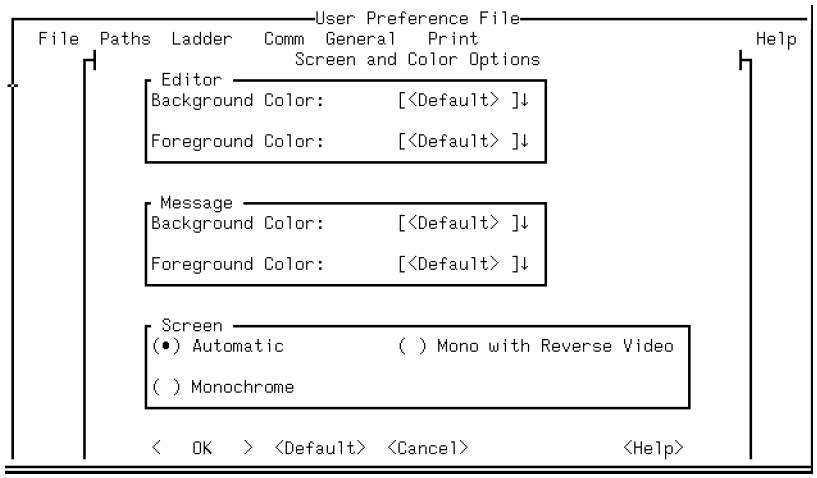


Figure 319 Screen and Color Preference Setting

The standard defaults are:

Editor - Blue background with White characters
Message - Grey background with Blue letters
Error - Red background with White letters

D.2.2.7 Print

This UPF Function is documented in the Printer Chapter.

D.3 User Level Settings

When MODSOFT is run, you can set a level indicator between the F7 and F8 by using the /L switch or setting the `user_level=` in Modsoft.UPF

In conjunction with the following table the setting you make when starting MODSOFT i.e:

MODSOFT /L3 (Level 3 for Example)

result in menus and functionality which reflects the selections. The levels are:

0 = Online Monitor (Read Only)
1 = Online allow change to registers
2 = PLC to Program
3 = Online with Start/Stop

4 = Program to PLC
5 = Full Online Program
6 = Full Offline
7 = Full Offline
8 = Full Offline and Executive Downloads

If the menu item is not reachable at a level, neither are its sub-contents

For consistency the user level default value of 8 is maintained throughout this document.

MODSOFT User Levels

Menu	Item	Minimum Offline Main Menu	Minimum Online
MAIN	Offline Combined Tools PLC to File File to PLC Transfer Options	6 6 6 2 4 6	n/a n/a n/a n/a n/a n/a
RDE	Enable All		1

Menu	Item	Minimum Offline Main Menu	Minimum Online
PLC OPS	Start Stop Single Sweep Constant Sweep Normal Optimize Clear Memory EE Prom Clock		3 3 3 3 3 3 5 1 1
LADDER	Objects Edit Network		5 5 5
NETWORK LIST	nsert Network		5
I/O MAP	Delete Drop Hold Up Time ASCII Port		5 5 5
CONFIGURATION	Port Default		5

Level 0 is an online Monitor mode. In this mode you can go Online and monitor the PLC program operation but can not alter the program contents or load files and work offline.

AppendixE
.ENV Files

- The .ENV file is created and used by the application. It's primary purpose is to direct the reloading of related program files.

Expert This Appendix provides the advanced user with internal technical detail regarding the structure of the .ENV file.

E.1 The Base File Set and .Extensions

A PLC program created by the application is made up of five (or more) base files referred to as the file set. The process of running the application through its various functions, in combination with user selectable information files, results in additional file extensions being created with the name originally assigned and directed to the location you previously set. The base file set is:

Filename.PRG	PLC Ladder logic
Filename.PCM	Network Comments for Ladder Logic
Filename.RFS	Reference Symbol File
Filename.CFG	Configuration file
Filename.RFD	Reference Data File (Created Offline)
Filename.ASC	ASCII file for PLC's that support ASCII messages.

E.1.1 The .ENV File

In contrast to pre 2.0 Modsoft applications Modsoft uses 1 file to load all the parts of a file set necessary to develop the program function.

This file has the extension .ENV and is an environment file used to load the programs. It also indicates the Modsoft directory these files are located in and uses stored parameters to establish your communications setup for this particular program.

Using the .ENV file allows you to load all the files associated with a program and set up the communications parameters one or two keystrokes on one menu instead of interacting with a number of menus and using many keystrokes.

Your applications program initial entry locates the function select cursor in the Select Program field of the Offline Pulldown. When you press the <**Return**> key, the .ENV files that exist in the PLC are listed in a window of File names on the display. If starting Modsoft for the first time there will be two filenames in the window.
These two files are supplied with the distribution set and are used for examples in this document. The names are "Time" and "Time_LL" When you pick one of these, all files related by the .ENV structure are loaded into the PC memory for execution.

You do not need to use the same directory as the the sample files supplied. There are two ways to establish directories of your choice:

1) Use the Offline Menu item Location of Program Files
2) Use the Modsoft.UPF as detailed in Appendix C.

i.e., env_directory = path where path is the drive and directory where the .env will exist.

E.1.1.1 Components of A .ENV File

The path and file name used below are consistent with the distribution software.

Environmental File Definition

File Content	Description
\MODSOFT\PROGRAMS\	Program Path
\MODSOFT\PROGRAMS\	Symbols Path
\MODSOFT\PROGRAMS\	Configuration Path
\MODSOFT\PROGRAMS\	Reference Data Path
\MODSOFT\PROGRAMS\	Macro Path
TIME	Program file name
TIME	Comment file name
TIME	Reference Symbol file name
TIME	Configuration file name
TIME	Reference Data file name
TIME	ASCII message file name
0	Unused
0	Modbus Mode
0	Communications Port
7	Baud rate
0	Parity
1	Data bits
0	Stop bits
1.0.0.0.0	Modbus Plus routing / Controller address
0	Modbus Plus adapter
1	Modbus 2 device
1.1	Ether Net address
111111	Auto Load vector 1 = load (each digit corresponds to one of the 6 files described above).
TIME.ENV	PlcName (this File)

These additional entries are included as of Modsoft 2.2 for custom printing:

```
[print_options]              {note: file section delimiter, no not delete}
    printer_index=2          index into printer device list
    device_file="modsoft.doc" Last file name or device used
```

configuration=N	configuration data list selection
segments="ALL"	Segments listing selection
sfc=Y	SFC printing enabled/disabled
sfc_elements="NONE"	selection of SFC elements to list
networks=N	Select to print/not print network lists
network_list="1"	select which networks to print
macro=N	Print macros Y/N
ascii="NONE "	Select ASCII to print
ref_values="NONE "	Select reference values to print
refs_used=N	Print used reference tables Y/N
ld=Y	do not touch
symbolic_info=Y	do not touch
global_cross_ref=NUMERIC	Select print of Global Cross References
operand_list=NUMERIC	Select print of Operand Lists
symbolic_cross_ref=Y	do not touch
local_cross_ref=N	Print local xrefs Y/N
single_network=Y	Print single network per page Y/N
header_file=""	Name of Listing Header file
header_1=""	Text for header line 1
header_2=""	Text for header line 2
header_3=""	Text for header line 3
footer_1=""	Text for Footer Line 1
footer_2=""	Text for Footer Line 2
footer_3=""	Text for Footer Line 3
parameters=N	Print the Doc Parameters Y/N
information=N	Print the Program Information Y/N
table_of_contents=N	Print Table of Contents Page Y/N
comment_location=Y	Put comments above node Y/N
coil_location=N	Put Coils to right of page Y/N
symbolic=N	

Note Internal Code tables are used to signify the Modbus Mode, Communications Port, Baud Rate, Parity, Data bits and Stop bits. You are advised to let Modsoft alter these fields through to PLC Ops Communications setup feature. ***Do Not edit these fields Yourself.***

This data sets up the directory path, file name, and controller specific information and can be seen by using DOS command TYPE or More or using an editor.

AppendixF
Miscellaneous

- This Appendix contains information regarding possible uses of the Modsoft application in more complex environments than the traditional DOS baseline.

F.1 Special Directory Consideration

If you want the current drive or directory for the runtime and temporary files (especially if using diskette drives), DOS environment variables have to be set accordingly. To define the source for the Modsoft internal runtime files (such as menus) the variable RT984 has to be set. To access the temporary path, where user changes are built without overwriting the original source file, the variable TMP984 is used. The default directory is \MODSOFT\RUNTIME for RT984 and \MODSOFT\TMP for TMP984.

F.1.1 SET CommandSummary (From the DOS level prompt)

Set RT984=C:\pathname of the Modsoft Runtime directory.

Set TMP984=C:\Pathname of the Modsoft Temporary directory.

Set SCREEN=MONO	Used for EGA, VGA, or CGA screens with monochrome monitors. Default is color, MONO for Monochrome screens.
Set POWER984=RV	Used to show reverse video powerflow for plasma screens. Default is Bold.
Set SWAP984=NO	No swap of Modsoft Lite to a Temp file when DOS is invoked.
=Pathname	If a pathname is given, Modsoft creates a swap file when entering DOS from Modsoft, and puts code in that file that has the effect of releasing 300K of RAM used by modsoft. If sufficient swap space is not found, no swap is performed, and the user will have about 240K of RAM to use in DOS. The default is to the current directory as the swap file directory.

F.2 Performance

There are a number of ways to improve the performance of Modsoft by increasing the amount of memory available for Modsoft to use. There is a direct relationship between Modsoft performance and available memory, up to a point of 6-8 Megabytes of RAM. Additional memory beyond that size will NOT improve performance.

The optimal configuration is:

1. DOS 5.0. Load DOS and all Device drivers HIGH.
   (The latest revision of Modbus + is recommended)
   Config.Sys DOS=HIGH
   DEVICEHIGH=C:\Bin\SA85.Sys

2. Run Smartdrive

   Config.Sys DEVICE=C:\DOS\SMARTDRV.SYS 2048

3. Configure extended memory

   Config.Sys DEVICE=C:\DOS\HIMEM.SYS
   Or expanded memory.
   Config.sys DEVICE=C:\DOS\EMM.Sys

4. Use the rest of memory as a Virtual Disk

   Config.sys DEVICEHIGH=C:\DOS\RAM
   DRIVE.Sys4096/E

   Autoexec.bat SET TMP984=Vdisk drive
   where Vdisk drive is a system specific
   value (i.e. it could be E:\ )
   Check your PC User Manual.

In addition to the above, any improvement in CPU speed helps because it executes Modsoft faster. With the slower machine, you may want to consider disabling the tracking help and/or using the /u switch to disable online used references.

F.3 Modsoft With Unicell PC

Modsoft may be operated in a Unicell environment where both Modsoft and Unicell applications are managed by DESQview Software.

F.3.1 Set up

The operational steps that follow assume that you are starting from a point where both Unicell and Modsoft have been properly installed and DESQview is fully operational.

Step 1 From the DOS prompt level type DV to start the DESQview application. This results in the display of a DESQview menu list.

Step 2 Select the “Open Window” item from the menu and observe the opening of a sub menu.

Step 3 Select “Add A Program” from the menu and observe the program add menu is displayed.

Step 4 Go to the entry for “Other” and Press the spacebar to bring up the entry line to specify the program path and name. for example you could enter: C:\MODSOFT.
This action results in a full screen display that allows you to specify data about the environment that the added program (Modsoft) is to run in.

Step 5 Fill in the data by:
Entering the Program Name (followed by a <Tab>)
Enter the two character ID which will alias the program name. For example you may pick MS for Modsoft.
Go to the Memory Size and change the default from 200K to 450K (recommended).
Set the PROGRAM =C:\Modsoft\Modsoft.exe <Tab>
PARAMETERS = See usage flags <Tab>
DIRECTORY = \Modsoft <Tab>
Leave the remaining option fields set to their default values.

The following figures illustrate the screen layout for the data entered in this step.

Step 6 Press <F1> To display the advanced options screen. Use the following figures as a guide to enter or verify the proper fields are defined.

Step 7 When step 6 is complete press the Return key and the screen returns to the “Add a Program” menu from which you select “Done”.

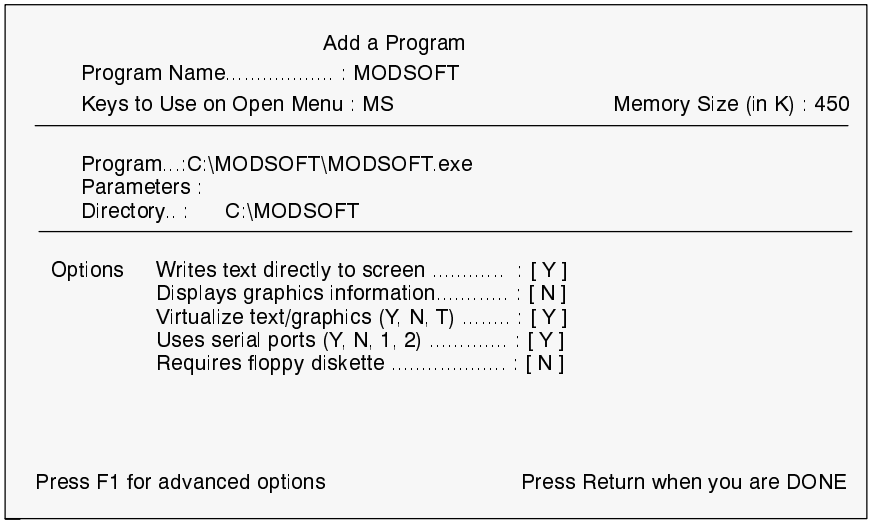


Figure 320 Example of Modsoft Parameter Entry Screen

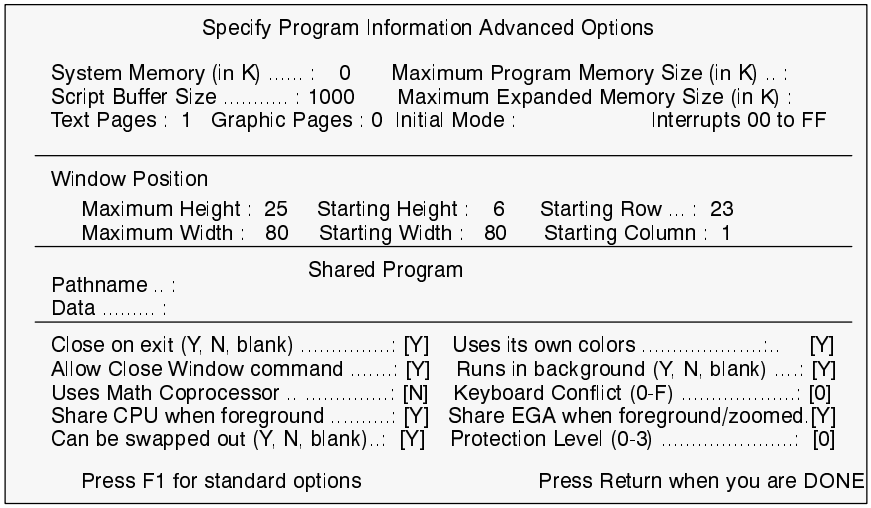


Figure 321 Example of Modsoft Advanced Option Entry

F.3.1.1 RUN

You can run Modsoft as a window application in concert with Unicell by performing the following steps:

Step 1 From DESQview select “Open window” menu, then select the Runtime Manager by entering the alias RU ↵. You can see the Unicell start-up cycle on the display.

Step 2 Press the <Alt> key to return to function to DESQview and select "Open Window" menu from which you pick Modsoft. The result is that Modsoft is presented in a window on the display and will execute in that window.

For Window size, placement etc., see the DESQview documentation.

☞ **Note** Your machine should have 2.5 Meg of available memory as a minimum.

F.4 Modsoft With Windows

F.4.1 Windows 3.0 and 3.1

You can run Modsoft in a windows environment with the Windows application program of 3.0 or 3.1. The following setup presumes Modsoft is already installed:

Tip Due to competing program interrupts, RTU communications is not a stable operating mode when using Modsoft in a Windows environment.

Step 1 Select the PIF editor from your `Main group` icon selection and fill in the displayed form as indicated.

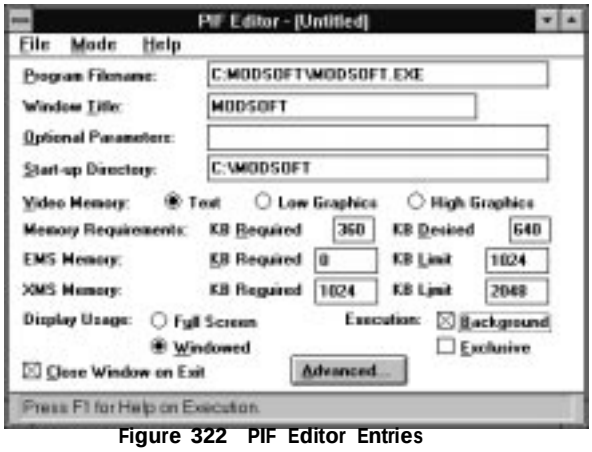


Figure 322 PIF Editor Entries

Step 2 Select the Advanced Options form and fill it in with the information you require using these examples.

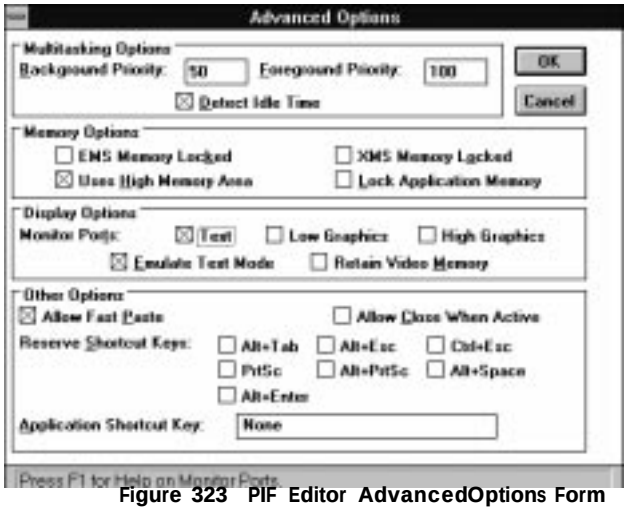


Figure 323 PIF Editor AdvancedOptions Form

After you have completed the Advanced Form you execute the File Save operation and assign the name to the file i.e.,MODSOFT.PIF.

Step 3 Click on the Icon that you just named (MODSOFT). Then with MODSOFT selected you can complete the file definition by filling in a new Program Item Properties form as illustrated below:

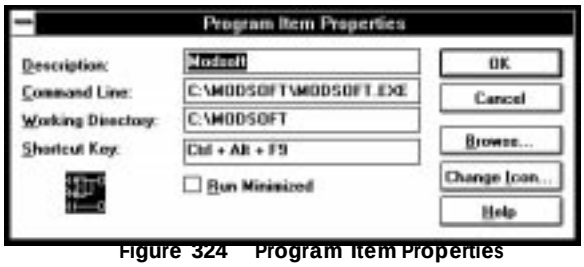


Figure 324 Program Item Properties

To execute Modsoft you double click on the Icon created with the description name above, for Modsoft, and a window will open with the Modsoft initial entry screen displayed. The Icon is available in the modsoft directory in a file called Modsoft.ICO

F.4.2 Windows 95

You can install and run Modsoft in a windows environment with the Windows 95 application program. The following presumes a new Modsoft installation:

Open a Dos window, insert disk 1 into drive A and at the Dos prompt type `A: Install`

Answer and or verify the series of questions prompted by the Modsoft install program.

Remove the first disk and put in the next as indicated by the prompt message on the display.

If the installation program finds an existing Modsoft directory it will halt with an appropriate message.

At the conclusion of the disk loading, you will find a `Modsoft` directory has been created.

Execute the program by typing “Modsoft” while in the Modsoft directory.

Alternatively you can use the Windows Explorer to list the contents of modsoft and create a shortcut for the Modsoft Application. If you choose this method you can drag and drop the shortcut Icon onto the desktop and then execute Modsoft from there by double clicking on the Icon.

F.4.2.1 Printing Modsoft from windows 95

Direct Connect If you are directly connected to a local printer you will experience an error using LPT1 from a windows 95 window. You may however, go to the `start` command and select `Shut Down Windows` → `Restart Windows in DOS Mode` then proceed with Modsoft.

To avoid the error and successfully print Modsoft from a window, you ***must*** remember to setup or **add the Modsoft printer type using the UPF file** to define the printer **including the device LPT1**

Networked To avoid the above error and print from a window, go to your network `printer` select the `properties` → `Details` then by `selecting Capture Printer Port` you can choose the Device LPT1 (or other DOS Device definitions) and add the network printer path. LPT1 then is routed to the Network Printer.

As stated above you ***must*** remember to setup or **add the Modsoft printer type using the UPF file** to define the printer **including the device LPT1 .** The UPF setup can not directly select network printers

F.4.3 Installing/Upgrading Modsoft Under Windows NT 4.0

These are the steps to install/update Modsoft under Windows NT 4.0.
(You cannot install Modsoft 2.51 and below under Windows NT 4.0 with these instructions.)

1. Create a temporary directory (in a DOS window) on the hard drive for the Modsoft Install/Update.

2. Copy Modsoft Install/Update disks (in sequence1,2 and 3) into this directory.

3. Create a shortcut to point to the install.exe or update.exe in this temporary directory.

4. Goto the shortcut's Properties -> Memory

5. Memory properties should be:

   Conventional memory
   Total: Auto
   Protected: Not checked
   Initial environment: Auto

   Expanded (EMS) memory
   Total: 4096

   Extended (XMS) memory
   Total: None
   Uses HMA: Checked

   MS-DOS protected-mode (DPMI) memory
   Total: Auto

F.4.4 Trouble Shooting Ideas

Installations with the Modsoft application in a windows environment can exhibit the following symptoms each of which has a probable solution:

Symptom - Modbus Plus communications some times stop or time out:
Check which version of Modbus Plus you are using. You should have or upgrade to Version 3.3 or higher. Contact your Modicon supplier or Customer service.

Symptom - Modbus RTU mode communications time out when running Modsoft in foreground (running in the “Active Window”):

If you are concurrently running any other application that is using Modbus Plus communications, be sure that it is Modbus Plus Version 3.3 or higher.

Symptom - Modbus RTU mode communications time-out when running Modsoft Lite in the background (running “Minimized”):
Use the PIF editor to increase Background Priority.

Symptom - Sluggish performance when using Modbus communications:

If available use Modbus Plus communications.

OR

Select the “Full Screen”, or “Lock Application Memory” option in the PIF editor. (These options affect the performance of other applications which may be running. Refer to the Windows User Guide.)

The memory window used by the MB+ card installed in your PC must be excluded from windows use. You can do this by:

Edit C:\windows\system.ini

Go to the section [386 Enh] and add the line

EMMexclude = xxxx - xxxx for example:

EMMexclude = D000 - D07F

F.4.5 Launching multiple Modsoft Windows

On occasion some users desire to have more than one modsoft setup. This method allows you run different applications. This is appropriate as long as certain guidelines are followed.

1. Each launch of Modsoft must have its own TMP directory, (if they are ever to be launched simultaneusly).

2. It is not necessary to have multiple copies of the Modsoft directory, as long as each launch has a separate supporting UPF and a separate TMP directory. The UPF can select TMP directory paths like c:\modsoft\tmp01 c:\modstmp2 etc.

3. It is not necessary to duplicate the Runtime and loadables directories, as they may be shared, and Modsoft never writes to these sets of files.

4. The DOS environment setup should not contain paths for the Runtime, Tmp or Programs directories. These should be controlled by the UPF.

5. It is necessary therefore to have separate batch startup files for selecting the correct setups for each launch.

6. An error is encountered if the same application batch is started from two different DOS windows.

7. Note that the VSL directory has a setup file that is used for TCP/IP comms. This is edited by the system manager, and may or may not be desired to be sharable.

An example of two modsoft launches:
Modsoft is installed in c:\modsoft\ as a standard installation.

Add to the modsoft directory additional tmp files as follows:
\modsoft\tmp1 \modsoft\tmp2
and optionallly program directories c:\modsoft\progs01 and c:\modsoft\progs02.

Create a UPF with the name app01.upf and set the paths as desired, including TMP to \modsoft\tmp1 with the runtime path being c:\modsoft\runtime and optionally, Program and names to c:\modosft\progs01

Create a UPF with the name app02.upf and set the paths as desired, including TMP to \modsoft\tmp2 with the runtime path being c:\modsoft\runtime and optionally, Program and names to c:\modosft\progs02

Create an application batch named app01.bat:
cd \modsoft
del modsoft.upf
copy app01.upf modsoft.upf
modsoft

Create a second application batch named app02.bat:
cd \modsoft
del modsoft.upf

```
copy app02.upf  modsoft.upf
modsoft

In DOS window one,  launch app01
In DOS window two,  launch app02
```

F.5 Mouse

Modsoft software supports mouse operations and offers User Preference File selection for convenient settings, as well as disable the mouse.

These commands can be either entered or put into an AUTOEXEC.BAT file

MOUSE

Each of the above takes parameters provided by the vendor-supplied manual for example, a MOUSE command with a parameter to indicate the mouse is to be used on COM2: rather than the COM1: default.

An AUTOEXEC.BAT file, for example, could look like:

```
Path = C:\;C:\DOS;C:\Mouse
MOUSE 2          (for com2 port spare)
```

once the appropriate drivers have been loaded, to interface Modsoft with the mouse type **modsoft** to start the application, and the mouse drivers look up whether the application has a mouse driver. If it does, mouse events are processed.

MOUSE Clicks

On a two button mouse, the buttons equate to:

Left Button = <**Enter**> key

Right Button = <**Tab**> key

Left Button Double Click = " * " key

MOUSE RELATED FILES

The following stored files are necessary when using the Logitech mouse:

Drivers

mouse.com Program that loads and activates the mouse TSR.

F.6 RTU Considerations

RTU protocol is designed for optimal speed. Transmission requirements do not allow for greater than a 3.5 character gap between characters. If this gap is exceeded the code assumes an “EOM” end of message has been received. Dedicated CPU’s like a PLC do not have the timing overhead that might be encountered in a PC environment where various interrupts have to be considered. A windows interface most likely increases the problem.

Some solutions to elevate this problem are:

1) Remove network drivers if possible.

2) Use a PC with 16550 style UART’s

3) Use the higher speed CPU

4) Use Modbus Plus transmission

5) Use ASCII protocol for RS-232 communications

F.7 Quantum Kernal Error Codes

Number of CPU RUN LED Blinks

SLOW (steady)	ERROR
0	Requested Kernel Mode
2	Kernel error
3	EXEC Checksum or EXEC not loaded
4	RAM Data Test
5	RAM Address Test
6	Hardware ModBus, MB+, ...
7	LMS Bus
8	Soft Errors

When the controller encounters an error, the main executive goes to the kernel area of the firmware to put the controller in a safe mode and enable the operator to determine the error type and corrective action to take. The following list shows the number of times the run LED blinks for each type of error and the crash codes for each. All codes are in hex.

NUMBER of BLINKS SLOW (steady)	CODE	ERROR
steady	0000	Requested Kernel Mode
2	80B	ram error during sizing
	80C	run output active failed
	82E	STACK ERROR IN MB CMD HANDLER
3	729	bus grant not recieved
	72A	not master asic on cpu
	72B	master config write bad
	72C	Quantum bus DPM write failure
	72F	plc asic loopback test asic/dpm
	730	plc asic BAD_DATA
4	604	UPI timeout error
	605	bad UPI response opcode
	606	upi bus diagnostic error
	607	modbus cmd-buffer overflow

	608	modbus cmd-length is zero
	609	modbus abort command error

NUMBER of BLINKS SLOW (steady)	CODE	ERROR
4 (cont)	614	mbp bus interface error
	615	bad mbp response opcode
	616	timeout waiting for map
	617	mbp out of syncronization
	618	mbp invalid path
	819	page 0 not paragraph aligned
	61E	bad external uart hardware
	61F	bad external uart interrupt
	620	bad receive comm state
	621	bad transmit comm state
	622	bad comm state trn_asc
	623	bad state trn_rtu
	624	bad comm state rcv_rtu
	625	bad comm state rcv_asc
	626	bad modbus state tmrO_evt
	627	bad modbus state trn-int
	628	bad modbus state rcv-int
	631	bad Interrupt
5	503	ram address test error
	52D	P.O.S.T. BAD MPU ERROR
6	402	ram data test error
7	300	EXEC not loaded
	301	EXEC Checksum
8	8001	Kernel errors
	8002	Kernel Prom checksum error
	8003	Unexpected executive return

Appendix G
Micro Configuration and I/O Map

- This appendix is provided so that users who might apply Modsoft with a Modicon Micro controller, but do not have the micro support documentation, will have this information for reference.

G.1 Micro Configuration

When configuring a Micro with a Hand Held Programmer (Modicon HHP) you have two devices designed in the same time frame and with the same *ease of use* philosophy. This coupling provides interaction with firmware features specific to the Micro product including use of new terminology. The Modsoft MMI maintains a traditional structure but incorporates as many new terms and operational characteristics as possible.

One notable exception in Micro Configuration is the Auto configuration possible with the HHP but not with Modsoft. In this chapter, you will see that for each Micro family model, there is a default screen that is the equivalent of the controller firmware default parameter values. You can therefore, download your initial configuration without editing the remaining configuration values once the PLC is defined.

Effort has been taken to store only legal configurations. You are prompted to complete the PLC configuration for a specific controller type in order to have a file that cannot accidently stop a machine. You can specify the parameters by filling in the fields or simply by choosing from a predefined set of values in a popup menu. Loading and saving a configuration is done through the "Offline- Select and / or Save functions" .

After selecting a ***listed*** program* from the main menu Offline function, the following programming operations are available from the Ladder editor programming menu:

Utility Commands Editors **Config** ASCII Save Quit

Your primary entrance to the configuration function, which includes the I/O Map, is via the **Config** pulldown.

* "Time" is provided on the distribution disk for use as an example.

All programming aspects of this menu are described in the programming Chapter. If you choose to create a new program name, Modsoft takes you directly to the Configuration Overview Menu. The Offline Configuration Menu offers these functions:

Utility OverView I/O Map Ports Segmnts Loadable Cfg Ext Quit

You can also access the Configuration function from Online and it's Tools menu but can not use the Overview function i.e.,:

Utility PlcOps I/O Map Ports Segmnts Loadable Cfg Ext Quit

When selected, the Configuration Overview screen is displayed and the Menu cursor is preset to

I/OMap

The Utility functions: Key Help, Program Information and Dos were explained in Chapter 3. The Reference Data and Symbol Table are discussed in chapter 6.

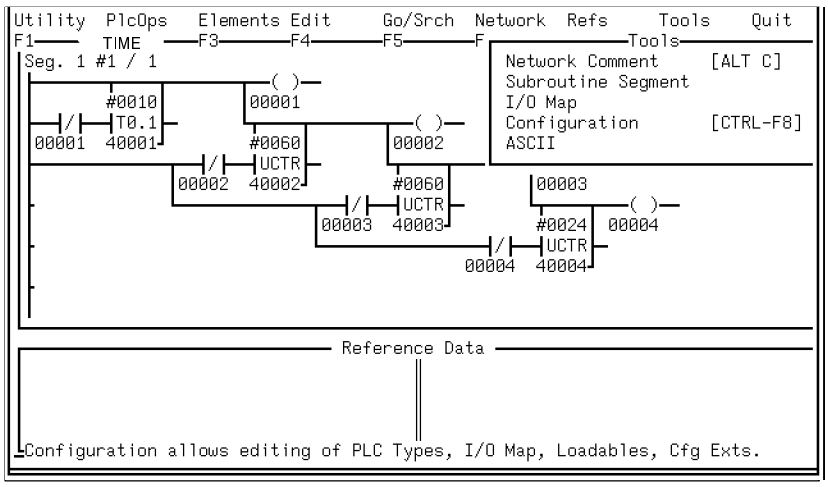


Figure 325 Online Tools Pulldown for Configuration Selection

G.2 Overview (Offline Only) for Micro PLC

The overview screen for the configuration file in the panel is always shown in the configuration screen. When the other editors of the configuration are called, they are laid over the overview screen. There are different sections on the overview, which can be accessed via entries on the menu function select line.

Within these sections, the cursor can be moved freely. Pressing any key starts editing. Pressing <**Esc** > during editing in a field quits the entry and restores the old value. Each field has set ranges to define valid entries. Entries are checked with their ranges when you intend to leave this field or by pressing <**Enter** >, <**Cursor Up**>, <**Cursor Down**>. Only if an entry is correct does the cursor leave the field. You can select from the following `Overview` screen submenu selections:

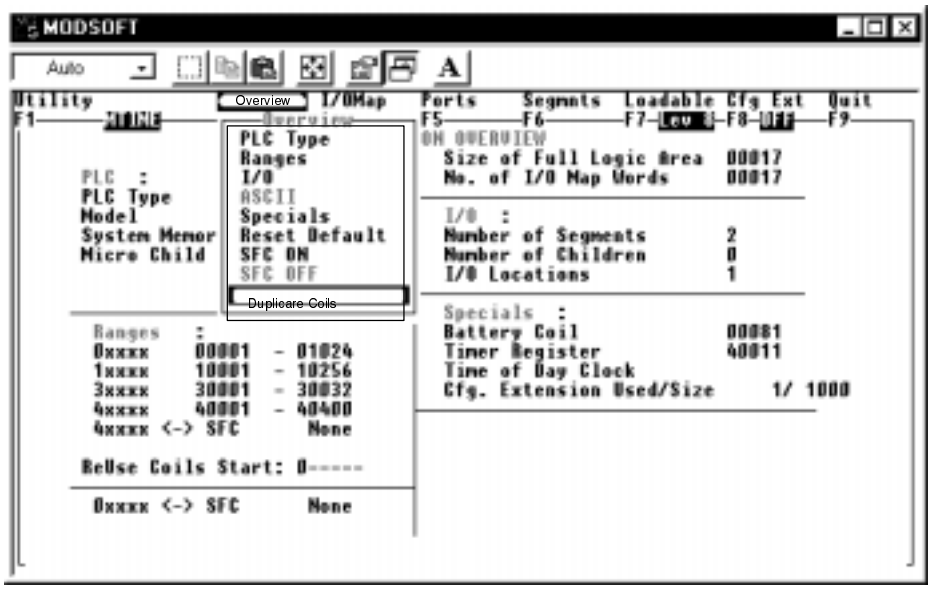


Figure 326 Overview Screen Example with OverView Menu Pulldown

G.2.1 PLC Type

This part of the screen sets up the style and memory configuration for a particular Micro programmable controller model. When starting up with a new system, this is automatically done by inserting default values without the need to select "Overview- PLC Type", because some parameter settings and allowed functions within the Ladder Diagram Editor depend on the selected controller type. A warning is shown if you change items in the PLC Menu, after chang-

ing parameters in the configuration. Also, there is a warning if parameters in the overview screen are changed before selecting the PLC type.

"PLC Type" sets the controller type for Modsoft. This list lets you select a specific Controller. You select the desired type using cursor up or down keys and the family type is displayed in a list for your selection.

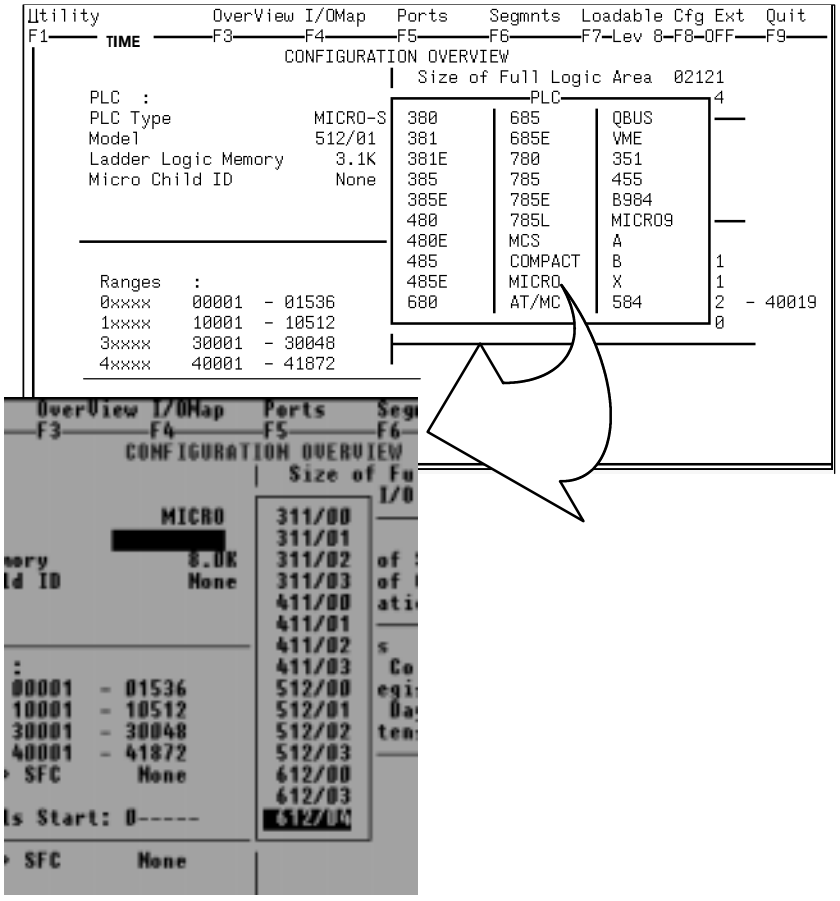


Figure 327 Example of controller types and Family Selection

For you to continue `PLC Type` parameter configuration, you need to know about the Micro *operating modes* available. To assist you in understanding these concepts, the drop connections (now called Expansion Links) are illustrated and defined. The physical relationships will help you do Model Configuration and when the configuration of I/O points is discussed later in this Chapter.

G.3 Micro Operating Modes

A Modicon Micro PLC can be configured to operate in one of three modes:

- *Single* mode—operating as a stand-alone programmable control system, managing its own (internal) fixed I/O resources (and, in the case of the 110CPU512 and 110CPU612 PLCs, able to manage external A120 I/O Modules).
- *Parent* mode—operating as the one PLC on a multi PLC link whose CPU can manage the fixed I/O resources of all the PLCs on that link.
- *Child* mode—operating as a PLC on an I/O expansion link, allowing some or all of its fixed I/O resources to be accessed and managed by the parent PLC on the link.

G.3.1 The Multi PLC Expansion Link

I/O *expansion* comprises a parent PLC and 1 ... 4 child PLCs *Linked* via standard six-position telephone cables. Each cable has an RJ11 connector on both ends. PLC-to-PLC connections are made at the RS-485 (**exp link**) port on each unit.

Only one PLC on the link can be configured as the parent. All other PLCs on the link must be configured as child PLCs. A PLC in single operating mode cannot be used on an expansion link.

Each child PLC is uniquely addressed with a `child ID #` in the range #1 ... #4. The fixed I/O resources of the child PLCs can be accessed and controlled by logic running in the parent.

☞ **Note** It is your responsibility as a user to make sure that each child PLC is given a unique child ID number. The child ID assignment is made by connecting the programming panel to the child and entering the number as part of the child's configuration.

Onboard I/O expansion is accomplished via serial, point-to-point connections between the parent and child PLCs, as follows.

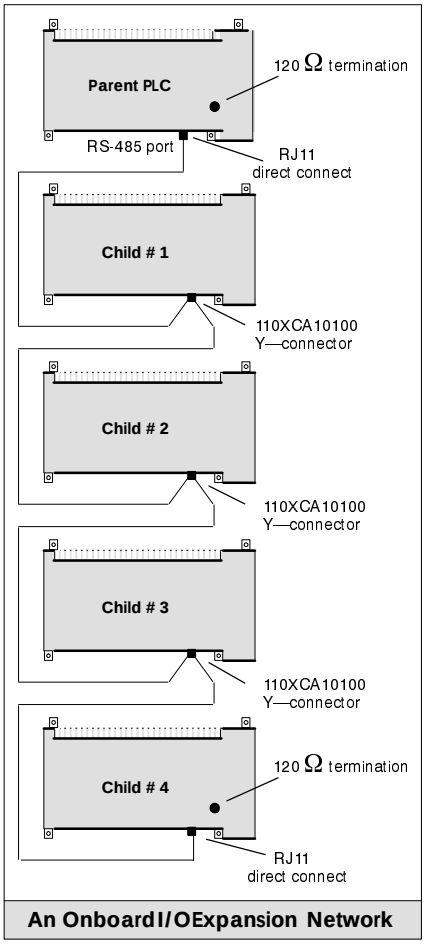


Figure 328 Expanding Onboard I/O Points

G.3.2 Using A120 External I/O Modules

110CPU512 and 110CPU612 PLCs are equipped with a 30-pin expansion port that allows the units to communicate with racks of A120 I/O. This port is dedicated to A120 I/O communications.

☞ **Note** 110CPU311 and 110CPU411 PLCs do not support A120 I/O expansion.

With A120 I/O expansion, 2 ... 4 racks are interconnected along a parallel address bus physically mounted on DIN rail. The PLC itself is always configured as rack 1, and the A120 I/O housing are configured as racks 2 ... 4.

A120 I/O expansion can be employed by the PLC in any of its three operating modes.

A120 I/O can be accessed only by the PLC to which it is connected. This means that the ladder logic program driving the A120 I/O and all the associated A120 I/O mapping must be stored in the PLC to which the the A120 I/O is connected.

> ☞ **Note** If a child PLC on a serial I/O expansion link uses A120 I/O expansion, the A120 I/O associated with that child cannot be accessed by the parent on the link. The child must be independently programmed with its own ladder logic, PLC configuration, and I/O map to handle that A120 I/O.

The following figure Illustrates the Mode and interconnections necessary to expand both serial and parallel I/O points.

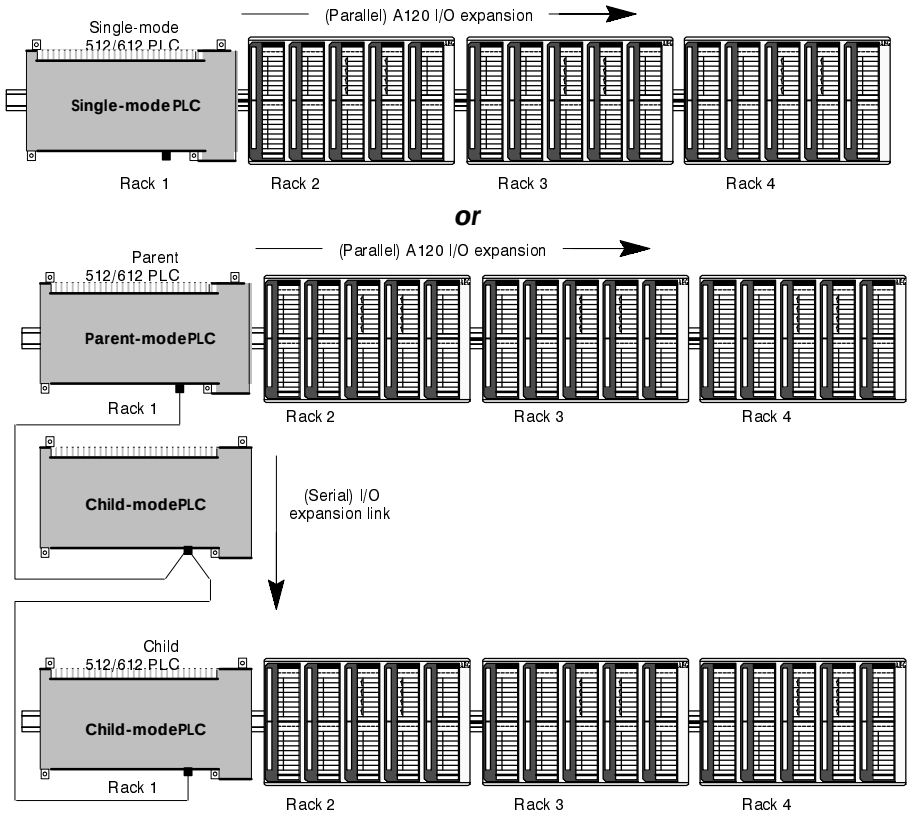


Figure 329 Operating Mode and External I/O Relationship

G.4 Configuration Parameters

Based on the PLC model type and PLC operating mode that you specify, the panel automatically configures the PLC with a full set of valid parameters. These *default configuration parameters* are shown in the following three tables.

G.4.1 Configuring a PLC in Single Operating Mode

If you configure a PLC in ***single*** operating mode, the display default configured parameters shown below are all you need to begin your ladder logic programming.

Autoconfiguration Parameters for a Single Mode Micro PLC

Parameter	110CPU Models	
	311 / 411	512 / 612
Number of 0*x* references	1024	1536
Number of 1*x* references	256	512
Number of 3*x* references	32	48
Number of 4*x* references	400	1872
Number of ladder logic segments	2 (the first for control logic and the second for subroutines)	2 (the first for control logic and the second for subroutines)
RS-232 port (**comm1**)	Dedicated Modbus mode: 8-bit RTU communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1	Dedicated Modbus mode: 8-bit RTU communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1
RS-232 port (**comm2**)	N/A	Dedicated Modbus mode: 8-bit RTU communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1
RS-485 port (**exp. link**)	Dedicated ASCII 8-bit ASCII communications, 9600 baud, even parity, 1 STOP bit	Dedicated ASCII 8-bit ASCII communications, 9600 baud, even parity, 1 STOP bit

The *Ports* Defaults, as defined in the above table, can be displayed under the PORTS selection on the configuration menu line as illustrated later in this Chapter.

G.4.1.1 Single Mode Modsoft Default Display

Using a Micro 512/01 for example, produces the following display which includes `I/O locations` and `Specials` default values:

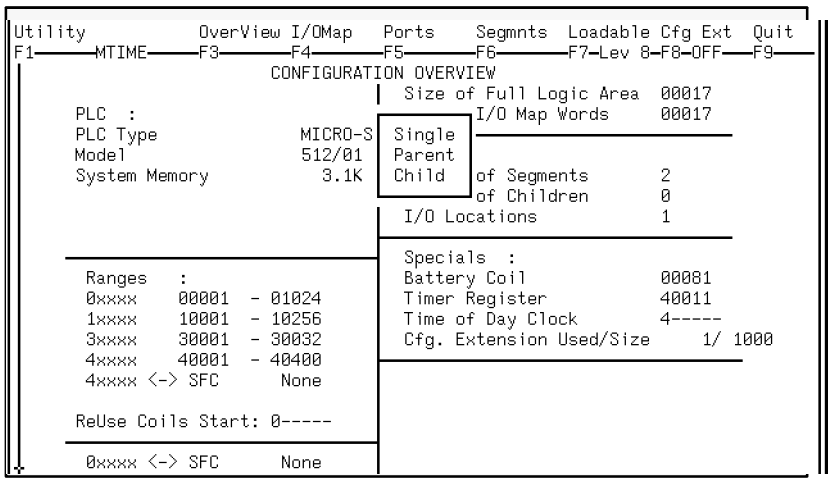


Figure 330 Modsoft Configuration Type Select option

G.4.2 Configuring a PLC in Parent Operating Mode

If you specify ***parent*** operating mode, you must specify the number of child PLCs that will be allowed on the I/O expansion link. The number must be in the range 1 ... 4.

Once you have specified this number, the PLC is ready to be programmed.

Autoconfiguration Parameters for a Parent Mode PLC

Parameter	110CPU Models	
	311 / 411	512 / 612
Number of 0*x* references	1024	1536
Number of 1*x* references	256	512
Number of 3*x* references	32	48
Number of 4*x* references	400	1872
Number of child PLCs on the I/O expansion link	*must be user-specified*	*must be user-specified*
Number of ladder logic segments	2 (the first for control logic and the second for subroutines)	2 (the first for control logic and the second for subroutines)
RS-232 port (**comm1**)	Modbus/ASCII toggling mode: 8-bit RTU/8-bit ASCII communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1	Dedicated Modbus mode: 8-bit RTU communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1
RS-232 port (**comm2**)	N/A	Modbus/ASCII toggling mode: 8-bit RTU/8-bit ASCII communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1
RS-485 port (**exp. net**)	I/O expansion network: 9-bit data communications, 125 ,000 baud, 1 STOP bit	I/O expansion network: 9-bit data communications, 125 ,000 baud, 1 STOP bit

The *Ports* Defaults, as defined in the above table, can be displayed under the PORTS selection on the configuration menu line as illustrated later in this Chapter.

G.4.2.1 Parent Mode Modsoft Default Display

Using a Micro 512/01 for example, produces the following display:

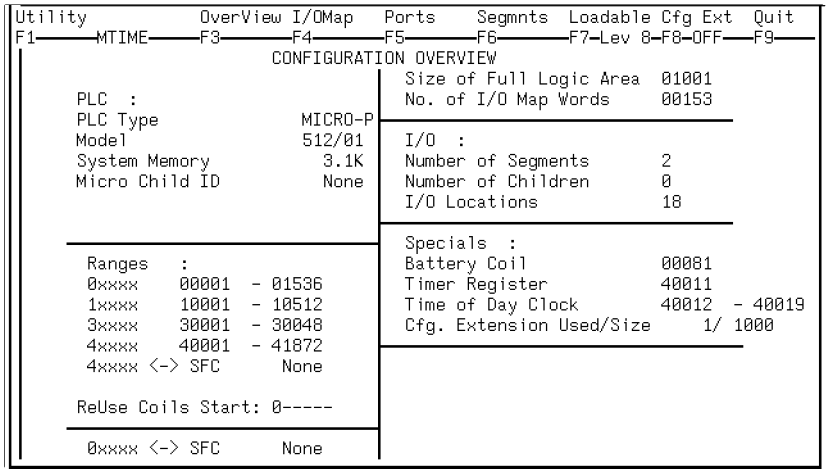


Figure 331 Configuration Of Parent Mode

G.4.3 Configuring a PLC in Child Operating Mode

If you specify ***child*** operating mode, you must assign a child ID number to the PLC. The number must be in the range 1 ... 4, and it must be unique to the particular child you are configuring with respect to all other child PLCs to be placed on the I/O expansion link.

Once you have specified the child ID #, the PLC is ready to be programmed.

Autoconfiguration Parameters for a Child Mode PLC		
	110 CPU Models	
Parameter	311 / 411	512 / 612
Number of 0*x* references	1024	1536
Number of 1*x* references	256	512
Number of 3*x* references	32	48
Number of 4*x* references	400	1872
Child ID #	*must be user-specified*	*must be user-specified*
Number of ladder logic segments	2 (the first for control logic and the second for subroutines)	2 (the first for control logic and the second for subroutines)
RS-232 port (**comm1**)	Modbus/ASCII toggling mode: 8-bit RTU/8-bit ASCII communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1	Dedicated Modbus mode: 8-bit RTU communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1
RS-232 port (**comm2**)	N/A	Modbus/ASCII toggling mode: 8-bit RTU/8-bit ASCII communications, 9600 baud, even parity, 1 STOP bit, Modbus address #1
RS-485 port (**exp. net**)	I/O expansion network: 9-bit data communications, 125 ,000 baud, 1 STOP bit	I/O expansion network: 9-bit data communications, 125 ,000 baud, 1 STOP bit

The *Ports* Defaults, as defined in the above table, can be displayed under the PORTS selection on the configuration menu line as illustrated later in this Chapter.

G.4.3.1 Child Mode Modsoft Default Display

Using a Micro 512/01 for example, produces the following display:

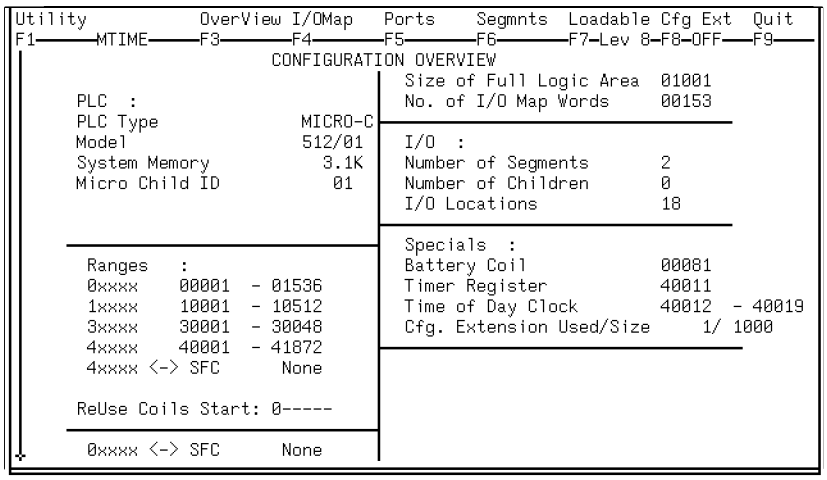


Figure 332 Configuration of Child Mode

G.5 Field Definitions for Configuration

G.5.1 PLC Operating Mode

The operating mode is described in the `PLC Type` entry in the top left data field of the screens. **`MICRO-S`** indicates single mode; **`MICRO-P`** indicates parent mode; and **`MICRO-C`** indicates child mode.

G.5.1.1 Model and Memory

984-311/00/01/02/03	Micro	2.1k
984-411/00/01/02/03,	Micro	2.1k
984-512/00/01/02/03,	Micro	3.1k
984-612/00/03,	Micro	3.1k
984-612/04	Micro	8.0K

Initialized defaults 311/00 Single

G.5.1.2 Child ID

The child ID # must be specified for a PLC that is configured in child operating mode. The Modsoft configuration defaults to an ID of 1. When you are configuring more than one child on an I/O expansion link, you need to make sure that each has a unique ID# in the range 1 ... 4.

This parameter does not apply to parent and single PLCs. For PLCs in either of these modes, the `Micro Child ID` is specified as `NONE`.

G.5.1.3 0*x*, 1*x*, 3*x*, and 4*x* Reference Ranges

The range of internal memory references is the same in all modes. The autoconfigured range assignments are the maximum number of references available for 110CPU51200 model.

Note The range of references is smaller for 110CPU311 and 110CPU411 models.

G.5.1.4 Number of Ladder Logic Segments

The autoconfigured number of ladder logic segments is 2. The first segment is available for normal control logic, and the second segment is available for subroutine logic.

G.5.1.5 Number of Child PLCs

If the PLC is configured in parent operating mode, you must specify the number of child PLCs that it can access on the I/O expansion link. The Modsoft configuration defaults to 1. If you want the ability to put more than one child on the link, change this parameter.

This parameter does not apply to single and child PLCs. For PLCs in either of these modes, the Number of Children is specified as 0.

G.5.1.6 I/O Locations

An I/O location is a unit of I/O associated with a particular type of Micro PLC. These I/O locations, which are described in more detail later in this chapter, include the fixed I/O built into the PLC and any A120 I/O modules connected to the PLC over the parallel expansion port.

☞ **Note** Only 110CPU512 and 110CPU612 models support A120 I/O; 110CPU311 and 110CPU411 models do not.

The 110CPU512 models all default to 18 I/O locations. This number allows you to support three or four fixed I/O locations—the discrete I/O, the high-speed inputs, and the generalized data transfer capability (more on these later)—as well as up to 15 slots of A120 I/O.

☞ **Note** 110CPU311 and 110CPU411 models will default to a much smaller number of I/O locations because these units do not support A120 I/O.

G.5.1.7 The Battery Coil

The operating system automatically sets aside reference 00081 as the battery coil. This coil operates much like the **batt low** LED on the PLC in that it turns ON when the optional battery needs to be replaced. You can tie this coil to an external alarm or display that warns you of the need for battery replacement.

When the battery coil goes ON, the battery should be replaced within 14 days.

G.5.1.8 The Timer Register

The operating system automatically sets aside output register 40011 as a free-running timer. This register is available to you for 10 millisecond applications in a ladder logic.

G.6 Addressing I/O Locations

The I/O map is a table in the PLC's system configuration memory that links reference numbers in the PLC's user data memory (0*x*, 1*x*, 3*x*, and 4*x*) to actual field inputs and outputs.

G.6.1 Fixed I/O Locations

A Modicon Micro PLC has five fixed I/O locations reserved for it in the I/O map editor.

- Location 1 for addressing fixed discrete input and output resources
- Location 2 for addressing counter/interrupt inputs
- Location 3 for addressing timer/counter inputs
- Location 4 for addressing fixed analog inputs and outputs
- Location 5 for addressing the transfer registers for a *generalized datatransfer* operation between a parent and child PLC

When you look at the I/O map in your panel software, the types of I/O points in each fixed I/O location are specified by an alphanumeric *locationtype* The table below shows the standard location types for the fixed resources on all models of Micro PLCs.

I/O Map Location Types for Fixed I/O			
I/O Location	**110CPU Model**	**Fixed Resources**	**Location Type**
Discrete (1)	31100, 41100 51200, 61200 61204	16 ... 24 VDC in / 12 relay out	MIC128
	31101, 41101, 51201	16 ... 115 VAC in / 8 triac out 4 relay out	MIC131
	31102, 41102, 51202	16 ... 230 VAC in / 8 triac out 4 relay out	MIC134
	31103, 41103, 51203, 61203	16 ... 24 VDC in / 12 FET out	MIC137
Counter / Interrupt (2)	All 411, 512, and 612 Models	8-bit counter/interrupt in	MIC140
Timer / Counter (3)	N/A in 311 Models	16-bit timer/Current count value	MIC147
Analog (4) All output channels have 12-bit resolution	612 Models only	4 in (0 ... 10, 12-bit), 2 out	MIC141
		4 in (1 ... 5, 12-bit), 2 out	MIC142
		4 in (+ 10, 12-bit), 2 out	MIC143
		4 in (0 ... 10, 15-bit), 2 out	MIC144
		4 in (1 ... 5, 14-bit), 2 out	MIC145
		4 in (+ 10, 16-bit), 2 out	MIC146
Generalized Data Transfer (5)	All Models	1 word in, 1 word out	MIC148
		2 words in, 2 words out	MIC149
		4 words in, 4 words out	MIC150
		8 words in, 8 words out	MIC151

Some of these locations may not be used for all PLC models—e.g., location 4 is reserved for fixed analog I/O which is available only in the 110CPU612s. When not used, a reserved fixed I/O location in the I/O map must be left empty—it cannot be used to address another type of I/O.

G.7 Addressing I/O on an Expansion Link

An I/O expansion link is created by daisy chaining up to five Micro PLCs together via cable connections at their RS-485 ports. One PLC must be configured as the parent, and the remaining units must be configured as child PLCs.

G.7.1 The Parent PLC

The parent PLC can address all its own fixed I/O resources as well as any fixed I/O resources residing in the child PLCs.

The fixed I/O locations of the parent PLC are automatically addressed for you. References for mapping additional I/O points from the parent are available as follows:

Physical Inputs	References (in User Data Memory)	Physical Outputs
	00001 ... 00012	Local fixed discrete outputs (12)
	00017 ... 00080	Reserved (A120 or child-based discretes)
	00081	Battery OK coil
Local fixed discrete inputs (16)	10001 ... 10016	
Reserved (A120 or child-based discretes)	10017 ... 10080	
Local interrupt/ counter inputs (8)	10081 ... 10088	
Reserved (child-based interrupt/timers)	10089 ... 10120	
Local timer/ counter input (1)	30001	
Reserved (child-based timers/counters)	30002 ... 30005	
Local fixed analog inputs (4)	30006 ... 30010	
Reserved (child-based analog inputs)	30011 ... 30030	
	40001 ... 40002	Local fixed analog outputs (2)
	40003 ... 40010	Reserved (child-based analog outputs)
	40011	10 ms timer
	40012 ... 40019	Time-of-day clock

G.7.2 A Child PLC

When you select child operating mode for a PLC, the ladder logic operating system assumes by default that all the fixed I/O points available on that PLC will be controlled by the parent on the network. Therefore, no values are assigned to the I/O map of a child PLC in its default state.

The fixed I/O locations in the child can be mapped in a screen associated with the parent's I/O map.

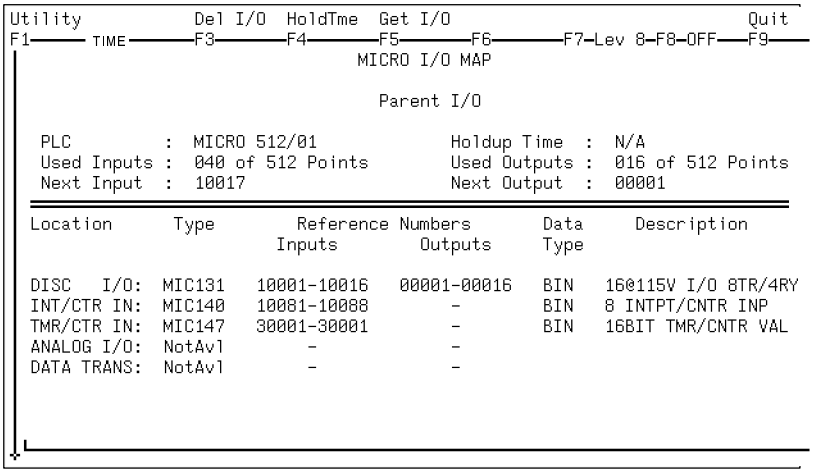
☞ **Note** Any A120 I/O connected to a child PLC must be addressed by the child. A120 I/O in a child cannot be accessed or controlled by the parent over the I/O expansion link.

G.7.3 An Example: An Expansion Link with all Fixed I/O Controlled by the Parent

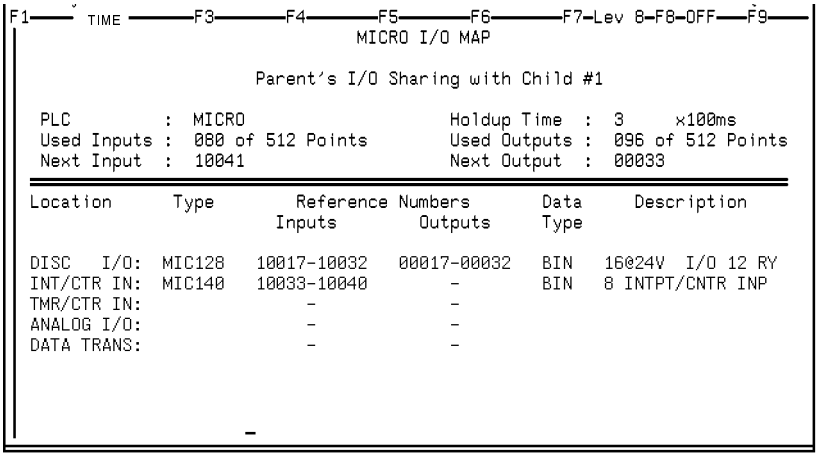
The system being configured in the following example consists of two 110CPU51201 PLCs, a parent and one child, on an I/O expansion link. The example shows three I/O map screens from Modsoft.

When you configure the parent, make sure that it is set for at least one child. The operating system will not allow the parent to access any of the child's I/O resources unless you have specified the existence of that child in the parent's configuration.

Screens 1 and 2 show the I/O maps for the fixed I/O locations in the parent and child that will be controlled by references in the parent's memory. Both I/O map screens are accessed while the programming panel is connected to the parent.



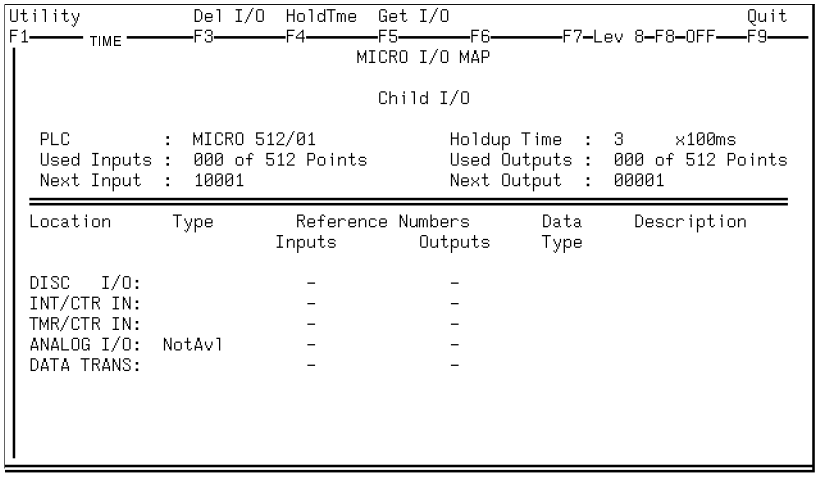
Screen 1. I/O Map for the Fixed I/O Points in the Parent



Screen 2. I/O Map for the Fixed I/O Points in the Child Accessed by the Parent

Notice that the location types used in the I/O map for the child place all the available fixed discrete input and relay output locations of the child under the control of the parent. **MIC128** maps all 16 of the child's 24 VDC inputs to references 10017 ... 10032 in the parent's memory and the 12 relay outputs to references 00017 ... 00032 in the parent's user data memory; **MIC140** maps the high-speed inputs to references 10033 ... 10040 in the parent's user data memory.

As a result, the I/O map screen that appears when the programming panel (see screen 3 below) is attached to the child shows no location types in it:



Screen 3. I/O Map for the Fixed I/O Points in the Child

G.8 Splitting Fixed I/O between Parent and Child PLCs

A child PLC has the option of splitting its fixed I/O resources with the parent—i.e., the child retains control over some of its own fixed I/O while the parent controls the rest. When fixed I/O resources are split, the I/O points controlled by the child must be addressed in the child's I/O map, and the I/O points controlled by the parent must be addressed in the parent's I/O map.

The key to splitting I/O is choosing the proper location types (see the table below) and placing them in the I/O map screens of the parent and child.

For example, if a child has 12 fixed FET outputs, you can I/O address one PLC's I/O map with a location type of **`MIC138`** (putting 8 FET outputs under its control) and the other I/O map with a location type of **`MIC139`** (putting the remaining four FET outputs under the other PLC's control).

I/O Map Location Types for Fixed I/O			
I/O Type		Location Type	110CPU Models
Discrete	16 ... 24 VDC in / 12 relay out	MIC128	31100, 41100, 51200, 61200 61204
	16 ... 24 VDC in / 8 relay out	MIC129	
	16 ... 24 VDC in / 4 relay out	MIC130	
	16 ... 115 VAC in / 8 triac out 4 relay out	MIC131	31101, 41101, 51201
	16 ... 115 VAC in / 8 triac out	MIC132	
	16 ... 115 VAC in / 4 relay out	MIC133	
	16 ... 230 VAC in / 8 triac out 4 relay out	MIC134	31102, 41102, 51202
	16 ... 230 VAC in / 8 triac out	MIC135	
	16 ... 230 VAC in / 4 relay out	MIC136	
	16 ... 24 VDC in / 12 FET out	MIC137	31103, 41103, 51203, 61203
	16 ... 24 VDC in / 8 FET out	MIC138	
	16 ... 24 VDC in / 4 FET out	MIC139	
Counter / Interrupt	8-bit counter/interrupt in	MIC140	All 512 & 612 Models
Analog (for 612 Models only) All output channels have 12-bit resolution	4 in (0 ... 10, 12-bit), 2 out	MIC141	61200, 61203
	4 in (1 ... 5, 12-bit), 2 out	MIC142	
	4 in (+ 10, 12-bit), 2 out	MIC143	
	4 in (0 ... 10, 15-bit), 2 out	MIC144	
	4 in (1 ... 5, 14-bit), 2 out	MIC145	
	4 in (+ 10), 2 out	MIC146	
Timer / Counter	16-bit timer/Current count value	MIC147	Default is NONE for all models
Generalized Data Transfer	1 word in, 1 word out	MIC148	
	2 words in, 2 words out	MIC149	
	4 words in, 4 words out	MIC150	
	8 words in, 8 words out	MIC151	

Both PLCs will read the same input data. Shared input data will not cause conflicts between the parent and child, and, therefore, the same fixed *inputs* can be mapped in both the parent and the child.

However, having both PLCs write the same output data can introduce errors. If the same outputs are mapped in both PLCs, the system will log an error against the parent, and it will be marked *unhealthy* in the PLC status table.

G.8.1 An Example: Splitting I/O

The following example shows two I/O map screens from Modsoft. They show how the 12 fixed relay outputs of a 110CPU51201 PLC configured as a child can be split between it and its parent.

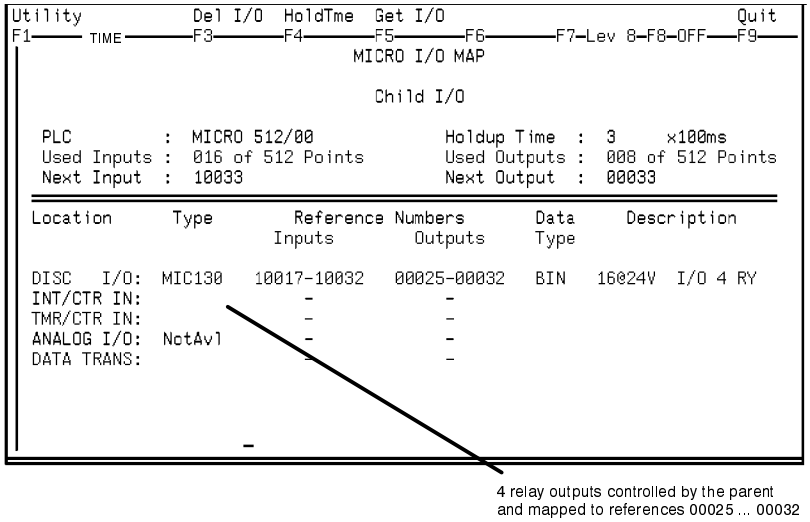
Screen 1 below is the map of the child I/O to be accessed by the parent. This I/O map screen is created while the programming panel is connected to the parent PLC. The location type for the discrete I/O is **MIC129**, indicating that the parent can access eight of the child's fixed relay outputs.

```
Utility          Del I/O  HoldTme  Get I/O                              Quit
F1 TIME          F3       F4       F5       F6       F7-Lev 8-F8-OFF  F9
                                MICRO I/O MAP

                        Parent's I/O Sharing with Child #1

 PLC         :  MICRO                  Holdup Time  :  3     x100ms
 Used Inputs :  056 of 512 Points      Used Outputs :  024 of 512 Points
 Next Input  :  10033                  Next Output  :  00025

Location      Type        Reference Numbers       Data     Description
                            Inputs       Outputs  Type

DISC   I/O:  MIC129   10017-10032   00017-00024   BIN   16024V  I/O 8 RY
INT/CTR IN:               -             -
TMR/CTR IN:               -             -
ANALOG I/O:               -             -
DATA TRANS:               -             -
```

8 relay outputs accessed by the parent and mapped to references 00017 ... 00024

Screen 1: Child I/O accessed by the parent

Screen 2 is the map of the child I/O that remains under the control of the child. This I/O map is created while the programming panel is connected to the child PLC. The location type for the discrete I/O is **MIC130**, indicating that the child maintains control over four of its fixed relay outputs.



Screen 2: Fixed I/Oresources controlled by the child

G.9 Addressing A120 I/O With Micro

110CPU512 or 110CPU612 models may use an optional A120 I/O expansion capability. When A120 I/O is used, it also needs to be I/O mapped in that PLC's system configuration memory.

You must edit the I/O map via panel software to address A120 I/O. Each A120 I/O module is assigned a location in the rack where it is housed.

Each physical rack connected to the PLC—*racks #2*, *#3* and *#4*—can have up to five I/O modules in it. As many as 20 A120 I/O modules (locations) can be addressed in a Micro PLC's I/O map. The first five locations are reserved for fixed I/O capabilities, and locations 06 ... 20 are for A120 I/O modules. The PLC reserves the following references for expanded I/O addressing:

- References 00017 ... 00080 for addressing discrete A120 output points
- References 10017 ... 10080 for addressing discrete A120 input points
- References 30002 ... 30005 and 30011 ... 30030 for addressing register/analog inputs from A120 I/O
- References 40003 ... 40010 are reserved for mapping register outputs from A120 I/O

☞ **Note** These reserved references may be used for addressing fixed I/O resources in other PLCs on an I/O expansion link if they are not used for A120 I/O addressing.

G.9.1 An Example: A Micro PLC with One Rack of A120 I/O

The following example uses two I/O map screens from Modsoft. The system being I/O mapped comprises a 110CPU51200 PLC and one rack of five A120 I/O modules—two BDAP212s and three BDAP216s.

The PLC uses only one of its discrete I/O points for this application. Therefore, a total of six I/O locations are used in this configuration—**`MIC128`** for the fixed I/O points, and five locations for the A120 I/O modules.

Screen 1 shows the I/O map for fixed I/O resources of the 110CPU51200 PLC. This PLC is considered rack #1 with respect to A120 I/O expansion. Note that only locations 1, 2, and 3 in rack #1 can be accessed.

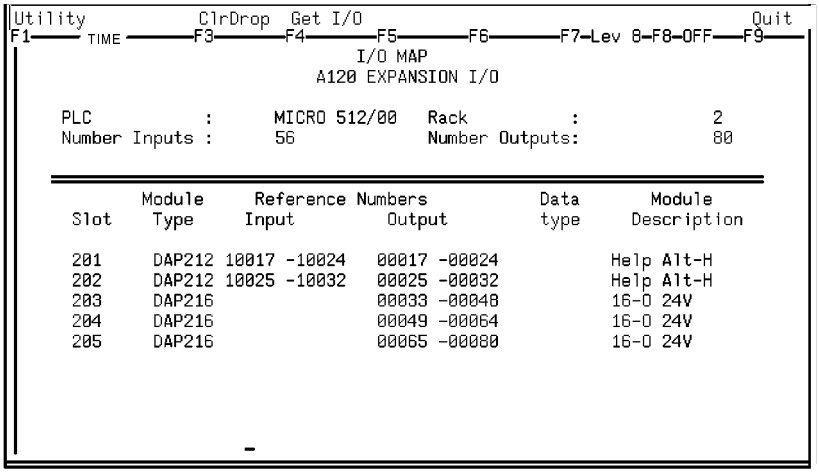
☞ **Note** In Modsoft , each rack is I/O mapped on a separate screen. You can move forward and backward through the screens—i.e., through the racks—by pushing <**PgUp**> and <**PgDn**>.

The A120 I/O in rack 2 is I/O addressed in the I/O map shown in screen 2. The A120 input points have been mapped to references 10017 ... 10032 and the output points to 00017 ... 00080 in the PLC's user data memory.

Altogether, this configuration uses 56 discrete inputs, 80 discrete outputs, and one counter/ timer register input.

```
Utility          Del I/O  HoldTme  Get I/O                          Quit
F1──── TIME──────F3──────F4──────F5──────F6──────F7-Lev 8-F8-OFF──F9────
                                MICRO I/O MAP

                                  Parent I/O

 PLC         :  MICRO 512/00            Holdup Time  :  N/A
 Used Inputs :  040 of 512 Points       Used Outputs :  016 of 512 Points
 Next Input  :  10017_                  Next Output  :  00017
═══════════════════════════════════════════════════════════════════════
Location      Type        Reference Numbers       Data     Description
                         Inputs       Outputs     Type

DISC   I/O:  MIC128   10001-10016   00001-00016   BIN   16@24V  I/O 12 RY
INT/CTR IN:  MIC140   10081-10088         -       BIN   8 INTPT/CNTR INP
TMR/CTR IN:  MIC147   30001-30001         -       BIN   16BIT TMR/CNTR VAL
ANALOG I/O:  NotAvl         -             -
DATA TRANS:  NotAvl         -             -
```

Screen 1. I/O Map for the Fixed I/O Locations (Rack 1)



Screen 2. I/O Map for A120 I/O Locations (Rack 2)

G.10 Starting The Micro

Now that the issues of Configuration and I/O Map editing have been presented you may wish to verify the operational capability of your controller.

If the operating system cannot find a valid configuration in the PLC's Flash or in its system configuration memory, it will power up as an *unconfigured* machine. A PLC will power up unconfigured the first time it is ever been started or when its configuration values have been cleared or corrupted.

You need to configure the PLC before you can write a logic program or service the I/O unless Autoconfiguring with a Hand held Programmer (see HHP documentation..

G.10.1 Configuring a Modicon Micro PLC

Step 1. Connect a programming panel, such as Modsoft or the HHP, to an RS-232 comm port on the PLC. Verify or set parameters using the Ofline-Change PLC Address Menu

Step 2. Using the panel's menuing system, go to the Configuration editor. (The path to the configuration editor will vary depending on the panel you are using, but it is a high-level screen that can be reached with minimal keystrokes.)

Step 3. Make sure that the panel knows which PLC model type (e.g., a 311/01, a 512/00 etc,) it is about to configure. The HHP displays his information automatically at startup; Modsoft prompts you to select the model type from a list.

Step 4. Select the desired operating mode for the PLC you want to configure. The operating mode can be either **s**ingle, **p**arent, or **c**hild.

Step 5. Transfer the configuration parameters from the panel (File) to the PLC.

Result. The panel configures the PLC with a full set of valid parameters based on the model and operating mode you specify. At this point, the PLC is configured. Verify by Starting the Controller and observing the controller Run light come on.

AppendixH
CodeBase Error Messages

- Modsoft file handling utilizing CodeBase functional implementation can produce and display errors given the appropriate condition.

H.1 Data Base Error Conditions

General Disk Access Errors

-10 Closing File

An Error occurred while attempting to close a file.

-20 Creating File

This error can be caused by specifying an illegal file name, attempting to create a file which is open, having a full directory, or by having a disk problem.

-30 Determining File Length

An error occurred while attempting to determine the length of a file.

-40 Setting File Length

An error occurred while setting the length of a file

-50 Locking File

An error occurred while trying to lock a file.

-60 Opening File

This problem is usually caused by specifying a file which does not exist. Another possible cause is an attempt to open more files than the operating system or compiler and handle.

Even though later versions of DOS do not limit you to 20 files, compiler library code often does. But first try increasing the FILES= in the CONFIG.SYS file

-70 Reading File

This could be caused by calling D4GO with a non-existent record number.

-80 Removing File

Error occurred while attempting to remove a file

-90 Renaming File

Error occurred while attempting to rename a file

-100 Seeking to File Position

Error occurred while attempting to position to a spot in a file

-110 Unlocking File

Error occurred while unlocking part of a file

-120 Writing to File

This error occurs when the disk is full.

DataFile SpecificErrors

-200 File is not a Data File

Error occurred while attempting to open a file as a .DBF file, if the file is not actually a true data file,. If it is a data file the header or data could be corrupted.

-210 Unrecognized Field Name

Function called with no corresponding field in data structure.

-220 Unrecognized Field Type

Type is not specified in the data file

-230 Record Length to large

Record length is longer than allowed

-300 Tag Entry Missing

A tag can not be located that corresponds to the record in this file.

-310 Not A Correct Index File

The file specified is not a true index file. Some internal Index inconsistency was detected.

-330 Tag Name Not Found

The tag name specified is not an actual tag as related to the current file.

-340 Unique Key Error

An attempt to add or create a file which would have resulted in a duplicate unique key tag.

Expression Evaluation Errors

-400 Comma Or Bracket Expected

-410 Expression Not Complete

-420 Data File Name Not Located

-422 IIF() needs Parameters of Same Length

-425 SUBSTR() and STR() need Constant Parameters

-430 Number of Parameters is wrong

-440 Overflow while Evaluating Expression

-450 Right Bracket Missing

-460 Sub-Expression Type is wrong

-470 Unrecognized Function

-480 Unrecognized Operator

-490 Unrecognized Value

-500 Unterminated String

Optimization Errors

-610 Optimization Error

A general CodeBase optimization error was discovered.

-620 Optimization Removal Error

An error occurred while suspending optimization.

-630 Optimization File Flushing Failure

An Error occurred during flushing of optimized file information.

RelationErrors

-710 Relation Error
A general CodeBase relation error was discovered.

-720 MatchingSlave Record Not Located

CodeBase could not locate master record corresponding slave record.

GeneralErrors

-910 UnexpectedInformation

CodeBase discovered an unexpected value in one of its internal variables.

-920 Out of Memory

CodeBase tried to allocate some memory from the heap, in order to complete a function call but was unsuccessful.

-930 UnexpectedParameter

Function passed an unexpected parameter value. This is often a null value.

-950 UnexpectedResult

A CodeBase function returned an unexpected value from another CB function.

AppendixI
TCP/IP Setup

□ This Appendix relates to the network parameter setup required to operate Modsoft in a TCP/IP communications mode and interface to a Network Option Module (NOE) using Ethernet protocol.

I.1 Example Install Files

You can automate the installation of the files and structure, presented in Paragraph 2, by using the supplied example `install` on your Modsoft distribution disk. You may use this example to modify your Autoexec computer configuration or modify the example to fit your installation (user name etc.) and implement it. Whatever your choice, the files described in Paragraph 2 are found in this directory and are configured as follows.

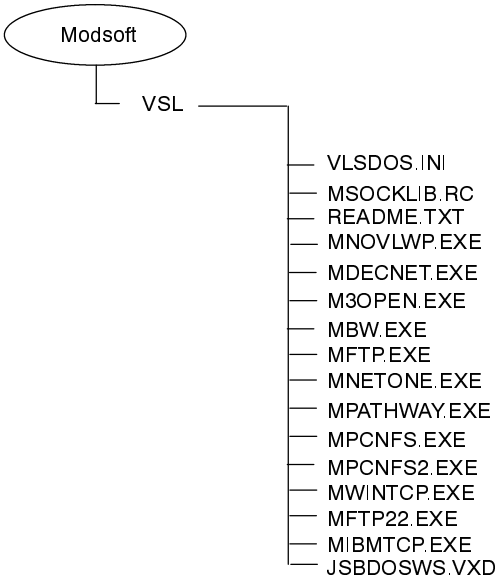


Figure 333 TCP/IP Install Disk Contents

I.2 Install Modsoft with TCP/IP

☞ **Note:** Please read and completely understand these instructions before installation. If a networking system administrator exists at your site, it is suggested that you have that person review these instructions before proceeding.

I.2.1 Overview

I.2.1.1 Modsoft Install for TCP/IP

Modsoft install for Transport Control Protocol/Internet Protocol (TCP/IP) access to a programmable logic controller.

- □ Install the DOS Terminate-and-stay-resident (TSR) based TCP/IP stack. (Note Windows 95 already has a stack installed)
- □ Install Modsoft from installation diskettes.
- □ Edit the vsldos.ini configuration file. For Win 95 edit the \windows\hosts and \windows\system.ini files.
- □ Set the VSL environment variable.
- □ Load Network and Network Module (netmod) drivers.
- □ Using Modsoft with TCP/IP

I.2.1.2 Changes for DECnet

Modsoft installation changes for DECnet access to a programmable logic controller.

- □ Set the VSL environment variable
- □ Load the DECnet Network Module (netmod)

☞ **Note:** Refer to paragraphs I.2.8 and I.2.9 for install changes specific to DECnet.

I.2.2 Install DOS Terminate-and-Stay-Resident(TSR) Based TCP/IP Stack (not necessary for Win95)

Follow the stack-provider's install-instructions. Modsoft 2.5 supports the following TCP/IP stacks:

DOS STACK	VSL Network Key
3Com 3+Open TCP v1.0, 1.1, 1.2, 2.0, 2.1	3open
Beame and Whiteside TCP/IP v2.2, 2.3, 3.0	bw
Digital Pathworks for DOS (TCP/IP) v4.1	decnet
D-Link TCP/IP for DOS v2.0	locus2
FTP PC/TCP v2.04, 2.05, 2.1	ftp
HP ARPA Services for DOS v2.1	3open
IBM TCP/IP	ibmtcp
FTP PC/TCP v2.2, 2.3	ftp22
Locus TCP/IP for DOS v2.0	locus2
Microsoft LAN Manager for MS-DOS (TCP/IP) v2.1 2.1a, 2.2	lanman
Novell LAN Workplace for DOS v4.0, 4.01, 4.1, 4.12	novlwp
Sun PC/NFS v3.0	pcnfs
Sun PC/NFS v3.5, 4.0a, 5.0, 5.1.5	pcnfs2
Ungermann-Bass Net/One TCP/IP v16.4, 16.5, 16.6	netone
Walker Richer and Quinn Reflection Network Series v2.1	3open
Wollongong Pathway Access v1.1, 1.1.1, 1.1.2, 1.2, 2.0, 3.0	pathway
Wollongong WIN/TCP for DOS v4.1.1	wintcp
Windows 95	winsock
Windows 95/Windows NT TCP/IP	****

**** Currently unavailable.

Note Windows NT have a 32-bit DLL-based TCP/IP stack. Until the netmod for Windows 95/Windows NT is available, the following procedure is necessary to use Modsoft TCP/IP communications in those environments.

Install the DOS, 16-bit TSR-based TCP/IP stack as per manufactures install-instructions.

De-install the Windows 95 TCP/IP stack, after installing the DOS TCP/IP stack. To de-install the Windows 95 TCP/IP stack, go into Control Panel → Networks and delete all applicable drivers. The adapter-card driver is necessary. i.e.- 3Com EtherLink III ISA (3C509b-TP) in PnP mode.

Tip: 32-Bit windows applications will not be able to communicate via TCP/IP once the 32-bit TCP/IP stack is replaced.

I.2.3 Install Modsoft from Installation Diskettes

Follow the installation procedures outlined in this Modsoft Programmer User Manual.

I.2.4 Edit the VSLDOS.INI Configuration File, and fot Win95, Edit the \Wndows\Hosts and \Wndows\System.ini files

The vsldos.ini file is located in the `<modsoft_installation_directory>\VSL directory`.

Each supported TCP/IP stack has a VSL network module (`netmod`) and corresponding `netkey`. Record that netkey in the vsldos.ini file, with your editor-of-choice.

In the vsldos.ini file, there is a heading entitled [vsl-default]. The first uncommented line following this heading (commented lines begin with a semi-colon) should read: `transport=.`

After editing the line that read: transport=, the line should read: `transport=<netkey>` where <netkey> is the VSL network key that corresponds to the installed TCP/IP stack. For example, if you are using Win95, then the first uncommented line following the [vsl-default] heading should read: transport=winsock.

A list of netkeys is supplied above, for your convenience, and also appears at the end of the `vsldos.ini` file.

For Win95, copy the \windows\hosts.sam file to \windows\hosts. Replace the first uncommented line in the file that reads:127.0.0.1 localhosts with <your_IP_address_in_dot_notation> <your_IP_alias> localhost.

For Example:

84.31.1.1 jdoe@inst.com localhost

For Win95 ad a line to the \windows\system.ini file under the [386Enh] section that reads:

device=<modsoft_installation_directory>\vsl\jsbdosws.vxe

Note The computer will have to be rebooted to have that change take effect.

I.2.5 Set The VSL Environment Variable

Modsoft 2.4 (and Up) TCP/IP communications requires the setting of the VSL environment variable. It is suggested that this environment variable be set in your `autoexec.bat` file.

Do: `set VSL=<modsoft_installation_directory>\vsl`

Expert: If you have installed Novell's Lan Workplace for DOS (LWP), there is a poorly documented environment variable, EXCELAN, that is required. It must be set to the installation directory of the LWP software.

I.2.6 Load Network And NETMOD Drivers (not necessary for win95)

Load all network drivers, in ISO-layered order. i.e.- physical, link, network, transport. Then load the network module (`netmod`).

The netmod is in the `<modsoft_installation_directory>\vsl.` Load it after the load of TCP/IP. The netmod is an executable, with the extension, .exe, having the same name as the network key BUT preceded by the letter, `m`.

For example: if you have installed Novell's Lan Workplace for DOS (LWP) the VSL network key (from above) is listed as `novlwp`, thus the VSL network module is `mnovlwp.exe.`

To load the netmod change to the <modsoft_installation_directory>\vsl.

Type: `mnovlwp.`

> **Note** It is suggested that the load of all drivers take place in a batch file, that is invoked on power-up of your computer.

An example follows, and assumes the LWP stack is installed:

lh LSL
lh 3C509
lh IPXODI
lh VLM
lh TCPIP
MNOVLWP

I.2.7 Using Modsoft With TCP/IP

To access a controller via TCP/IP in Modsoft, select `TCP/IP` under `Protocol` from either the `Offline→Change Plc Address` sub-menu or the `Online→Direct to Plc` sub-menu. (TCP/IP access is also available from the `Utility→Quantum Backplane→Direct to Plc` sub-menu and the `Utility→PLC & IO Status→Direct to PLC sub-menu`.)

The `Dest_Port` should remain at 502, the default. The port, 502, is Schneider Automation's well-known TCP/IP port and would only change if testing to a firewall.

The `Dest_Index` should remain at 0, the default. A destination index of 0 accesses the PLC at Head 0.

Input the IP address (symbolically or in dot notation) of the Network Options Ethernet (NOE) module in the TCP/IP Address field.

☞ **Note** the IP address (or the translated symbolic address) must agree with the IP address in the configuration extension of the controller you want to access. ***It is essential that the configuration extension be correctly set-up in order to access the NOE module via TCP/IP.*** See the Modicon Quantum Ethernet TCP/IP Module User Guide for details.

☞ **Note** If you enter a TCP/IP address that is a broadcast or multicast address, or enter an address that is illegal, the cursor is placed at the beginning of the TCP/IP Address field and Modsoft waits for you to re-enter the correct address.

I.2.8 Set the VSL Environment Variable

the Virtual Socket Library (VSL) is used in Modsoft 2.5 (and beyond) to provide access to a PLC via DECnet. VSL requires setting the VSL environment variable. It is suggested that this environment variable be set in your autoexec.bat file. type:

```
set VSL=modsoft_installation_directory\vsl
```

I.2.9 Load the DECnet Network Module (netmod)

Load all the DECnet drivers then load the network Module (netmod), mdecnet.exe. you will find the netmod file in the modsoft_installation_directory\vsl.

To load the netmod change to the modsoft_installation_directory\vsl, type:

```
mdecnet
```

The example that follows assumes the loading of decnet drivers delivered with BGAT:

```
1h dllndis
1h netbind
1h sch/h
1h dnnethat
mdecnet
```

I.3 Example Network Topology

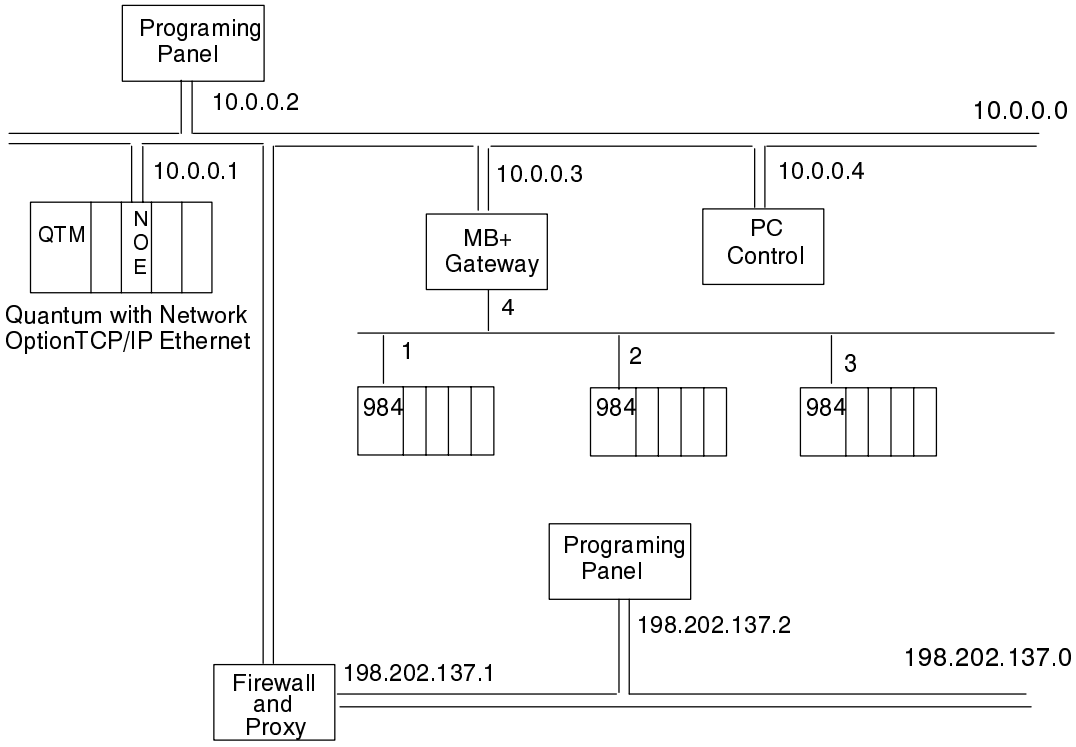


Figure 334 Example Networked Product Mix

I.4 Common TCP/IP Error Messages

When first configuring and attempting to use Network Option Ethernet modules for Ethernet (NOE), software error detection and the resulting messages may include:

TCP/IP Can't assign requested address

If you see this message, ensure that a TCP directory exists. The TCP directory should contain `services, protocol, host` and `resolv.cfg` file.

Timeout without a PLC response - review PLC Address Setup

If you see this message, check the communications_timeout value in the UPF file and increase, if necessary. A value of 30 seconds should be sufficient for TCP/IP.

The Selected Communications mode is not available

If you see this message, ensure that you can `PING` the node in question. Verify that the TCP/IP and Network module drivers are loaded. Verify that the `vlsdos.ini` file correctly reflects the Network Module (netmod) which corresponds to your TCP/IP stack. The Modsoft Read.Me can assist you with information on loading the communication drivers, and configuring the `VLSDOS.INI` file.

Symbols

Numbers

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

Y

Z