AutoMax®
Programming Executive
Version 3.8

(M/N 57C600)
(M/N 57C602)
(M/N 57C620)
(M/N 57C622)
(M/N 57C650)

Instruction Manual J2-3106-1

RELIANCE ELECTRIC

The information in this user's manual is subject to change without notice.

DANGER

ONLY QUALIFIED ELECTRICAL PERSONNEL FAMILIAR WITH THE CONSTRUCTION AND OPERATION OF THIS EQUIPMENT AND THE HAZARDS INVOLVED SHOULD INSTALL, ADJUST, OPERATE, OR SERVICE THIS EQUIPMENT. READ AND UNDERSTAND THIS MANUAL AND OTHER APPLICABLE MANUALS IN THEIR ENTIRETY BEFORE PROCEEDING. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN SEVERE BODILY INJURY OR LOSS OF LIFE.

WARNING

THE USER MUST PROVIDE AN EXTERNAL, HARDWIRED EMERGENCY STOP CIRCUIT OUTSIDE THE CONTROLLER CIRCUITRY. THIS CIRCUIT MUST DISABLE THE SYSTEM IN CASE OF IMPROPER OPERATION. UNCONTROLLED MACHINE OPERATION MAY RESULT IF THIS PROCEDURE IS NOT FOLLOWED. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.

Table of Contents

Appendices

List of Figures

List of Tables

1.0 INTRODUCTION

The products described in this manual are manufactured or distributed by Reliance Electric Industrial Company.

Version 3.x of the AutoMax Programming Executive uses Microsoft Windows™ to provide a graphic interface for offline operations such as mapping variable names to I/O points and creating a hierarchical structure of systems, sections and racks for each application. The online operations in AutoMax Executive V3.x, e.g., monitoring variables, are not run under Windows.

AutoMax Executive V3.x will run on IBM-compatible 80386-based (or higher) systems having at least 2 MB of RAM and a 40 MB hard drive. Significant improvement in performance is possible with additional memory capacity. The operating system for your computer must be version 5.0 (or later) of MS-DOS™ (or equivalent).

AutoMax Executive V3.x will run under Windows Version 3.1 (or later). It will not run under earlier versions of Windows. You must use Windows 386 enhanced mode. AutoMax Executive V3.x will not run in Windows real mode. The current version of Windows can be purchased from Reliance (order M/N 57C389) or from Microsoft. Microsoft can also provide upgrades if you have an earlier version of Windows.

Local Area Network Features

AutoMax Executive V3.x can be used on a workstation connected to a local area network. Network facilities (tokens and file locking, Version Control Library) are included in this release of AutoMax Executive V3.x. If more than one person has access to files for your application, we recommend that you use these facilities to prevent corruption of your database files and loss of changes to application task files. See appendix N for information on Tokens and Locking. See appendix P for information on using the Version Control Library functions.

Note that the Version Control Library (VCL) facility requires the Intersolv™ Polytron Version Control System (PVCS) Version 5.0 (or later). The VCL functions are not compatible with earlier versions of PVCS. PVCS is not required to use the AutoMax Programming Executive and is not distributed as part of the Programming Executive. However, you must have PVCS installed on your network drive in order to use the VCL features in AutoMax Executive V3.x.

1.1 Manual Contents

Listed below is a brief section-by-section summary of the contents of this instruction manual. Because this instruction manual covers a wide range of information that applies specifically to the AutoMax Executive V3.x Programming Executive and to the AutoMax distributed control system in general, you may want to go over this summary carefully to decide which sections are of specific interest to you depending upon your prior knowledge of the AutoMax distributed control system and your knowledge of Windows.

Appendix J, a quickstart guide, may be helpful as a brief overview of the Version 3.x Programming Executive.

- Section 2.0 - Introduction to the AutoMax Distributed Control System - Contains a general description of the AutoMax distributed control system and an overview of the AutoMax Programming Executive.
- Section 3.0 - Installing the AutoMax Software - Provides instructions for installing the AutoMax Programming Executive software and the Windows software. Also provides instructions for installing a mouse.
- Section 4.0 - Getting Around in Windows and AutoMax - Describes Windows functionality as it relates to AutoMax V3, Programming Executive file types and storage, and the text editor and PC editor used by AutoMax.
- Section 5.0 - Using the System Configurator - Describes the System Configurator, which is used to organize systems, sections, and racks. Contains procedures for executing commands from the system, section, and rack menus; procedures for loading the operating system to the rack; procedures for importing tasks created with previous versions of AutoMax and ASD; and procedures for setting up your Windows and AutoMax software.
- Section 6.0 - Configuring Racks - Describes the Rack Configurator, which is used to add modules to a rack and work with modules in existing racks.
- Section 7.0 - Configuring I/O and Common Memory Locations - Explains how to configure I/O points on a module.
- Section 8.0 - Using the Task Manager - Describes procedures for creating and working with application tasks. Also describes how to access online menu functions.
- Section 9.0 - System Security - Describes online security features and methods for setting and changing passwords.
- Section 10.0 - AutoMax Processor Overview - Describes the AutoMax Processor module and online operation of the AutoMax distributed control system.
- Section 11.0 - AutoMax ON LINE Menu - Describes how to load, run, stop, monitor, and modify application tasks in the rack. Also describes AutoMax and Universal Drive Controller Processor error logs.
- Section 12.0 - ON LINE Menu: Connect - Describes how to change the baud rate, enter/release the password, and enable and disable AUTO RUN of application tasks.
- Section 13.0 - ON LINE Menu: Info/Log - Describes how to display information about the system software on any AutoMax Processor or Universal Drive Controller in the rack and how to view the status and error log for tasks.
- Section 14.0 - ON LINE Menu: Transfer - Describes procedures for changing the default path or viewing the contents of the default system. Also contains procedures for loading tasks onto the rack and saving application tasks from the rack.
- Section 15.0 - ON LINE Menu: Running Tasks - Describes how to run tasks.
- Section 16.0 - ON LINE Menu: Stopping Tasks - Describes how to stop tasks.

- Section 17.0 - ON LINE Menu: Deleting Tasks - Describes how to delete tasks from the rack.
- Section 18.0 - ON LINE Menu: Monitoring and Editing Tasks - Describes how to monitor and change status of variables and I/O points, display and modify Ladder Logic sequences in real time, and force and un-force variables.
- Section 19.0 - ON LINE Menu: Error Clear - Describes how to clear the Processor or UDC error log or errors displayed on the Processor LEDs.
- Section 20.0 - Software Troubleshooting - Describes typical errors that can occur and procedures for correcting these errors.

Appendix A	Processor Module Ports
Appendix B	Processor Module Error and Status Codes
Appendix C	AutoMax Task Execution
Appendix D	Limitations on Task Size
Appendix E	Windows Command Summary
Appendix F	Database Definitions
Appendix G	Importing Racks from AutoMax Systems Containing Network Variable (.NET) Files
Appendix H	Using Foreign Modules in an AutoMax Rack
Appendix I	Transferring Tasks created with AutoMax V3 to ASD
Appendix J	Quickstart Guide
Appendix K	Import Notes
Appendix L	AutoMax Bill of Material Generator
Appendix M	Copying Systems and Racks
Appendix N	Tokens and Locking
Appendix P	Version Control Library
Appendix Q	Version Update Utility
Appendix R	New Features in This Release
Appendix S	Part Numbers of UDC Operating System Versions
Appendix T	What Can Go Wrong When Loading Tasks or Files

1.2 Additional Information

The Resource AutoMax Executive software incorporates the Norton Editor and a version of Kermit communication software. The operation of these two software packages within AutoMax is transparent to the user. In order to avoid duplication, this manual will refer the user to the instruction manuals describing the Norton Editor (J-3618) and Kermit Software (J-3616). If you will be using these two software packages, it is important that you become familiar with them as you learn about AutoMax.

Information about personal computers in this manual is general enough to apply to most of the personal computers that can be used to run the AutoMax Executive Software. For any specific questions about your personal computer, the DOS operating system, or the printer you are using, refer to the documentation that came with your personal computer or printer.

You must be familiar with all the instruction manuals that describe your system configuration. This may include, but is not limited to, the following:

- S-3000 SD3000/SF3000 Drive (Binder)
- S-3001 SA3000 Drive (Binder)
- S-3053 SA3100 Drive (Binder)
- S-3002 SA500 Drive (Binder)
- J-3616 KERMIT COMMUNICATIONS SOFTWARE INSTRUCTION MANUAL
- J-3618 NORTON EDITOR INSTRUCTION MANUAL
- J-3636 COMMON MEMORY MODULE INSTRUCTION MANUAL
- J-3650 AutoMax PROCESSOR MODULE INSTRUCTION MANUAL
- J-3669 AutoMax Pocket Reference
- J-3675 AutoMax ENHANCED BASIC LANGUAGE INSTRUCTION MANUAL
- J-3676 AutoMax CONTROL BLOCK LANGUAGE INSTRUCTION MANUAL
- J-3677 AutoMax LADDER LOGIC LANGUAGE INSTRUCTION MANUAL
- J2-3107 ReSource AutoMax and DPS SOFTWARE LOADING INSTRUCTIONS
- IEEE 518 GUIDE FOR THE INSTALLATION OF ELECTRICAL EQUIPMENT TO MINIMIZE ELECTRICAL NOISE INPUTS TO CONTROLLERS
- Microsoft Windows User's Guide
- Your personal computer and DOS operating system manual(s)
- Other instruction manuals applicable to your hardware configuration

1.3 Related Hardware and Software

This manual includes information about the following hardware and software:

- ReSource AutoMax Executive Software Version 3.x (M/N 57C395 and 57C397). This software contains both the Executive and the operating systems to be loaded onto AutoMax Processors and UDC modules in the rack.
- AutoMax Processor Module (M/N 57C430A and later)
- Microsoft Windows 3.x (M/N 57C389)
- IBM-compatible 80386-based personal computers used to run the AutoMax Executive Software
- Mouse pointing tools

Note that a mouse is not required to use AutoMax Executive V3.x. While many functions will be easier to use with a mouse, all of the functions in AutoMax Executive V3.x are accessible using only the keyboard.

Distributed Power Drives

Distributed Power System (DPS) drives are described in other documentation. See section 1.2 for the part numbers of the documentation.

1.4 Compatibility with Processors and Earlier Versions of the Programming Executive

Version 3.x of the AutoMax Programming Executive software requires AutoMax Processor module M/N 57C430A, 57C431, or 57C435. M/N 57C430 and 57C430A (and later) cannot co-exist in the same rack. Version 2 software can co-exist on the same personal computer with Version 3.x Executive software.

1.4.1 Programming Executive and Operating System

The following table shows compatibility between versions of the Programming Executive and the operating system in the AutoMax rack.

Table 1.1 - AutoMax Programming Executive and Operating System Compatibility

Version of AutoMax Programming Executive	Operating System in Rack									
	before V2.1B	V2.1B or Later	3.0 A	3.1A to 3.2B	3.3A to 3.3J	3.4	3.5	3.6	3.7	3.8
before V2.1B	yes	no	no	no	no	no	no	no	no	no
V2.1B or later	yes*	yes	no	no	no	no	no	no	no	no
V3.0A	yes*	yes*	yes	no	no	no	no	no	no	no
V3.1A to V3.2B	yes*	yes*	yes*	yes	no	no	no	no	no	no
V3.3A to V3.3E	no	no	no	no	yes	no	no	no	no	no
V3.3F to V3.3J	yes*	yes*	yes*	yes*	yes*	no	no	no	no	no
V3.4	yes*	yes*	yes*	yes*	yes*	yes	no	no	no	no
V3.5	yes*	yes*	yes*	yes*	yes*	yes	yes	no	no	no
V3.6	yes*	yes*	yes*	yes*	yes*	yes	yes	yes	no	no
V3.7	yes*	yes*	yes*	yes*	yes*	yes	yes	yes	yes	no
V3.8	yes*	yes*	yes*	yes*	yes*	yes	yes	yes	yes	yes

* Limited to the functions available in the operating system. For example, 7010 Processors are not supported when using the V3.1A Programming Executive with the V3.0 operating system.

1.4.2 Programming Executive Versions and Part Numbers

The following table shows the relationship between the operating system part numbers which are displayed in the online Info/Log and the Programming Executive versions. See appendix S for part numbers for the UDC operating system.

Table 1.2 - Part Numbers for Versions of the AutoMax Operating System

Part Number	OS Type	Programming Exec. Versions
422503-011B	6010 Standard	V1.0A
422503-011C	6010 Standard	V1.0B
422503-011D	6010 Standard	V1.0C
422503-020A	6010 Standard	V2.0A
422503-020B	6010 Standard	V2.0B
422503-021	6010 Standard	V2.1, V3.0A
422503-021A	6010 Standard	V2.1A
422503-021B	6010 Standard	V2.1B, V3.1A
422503-021C	6010 Standard	V2.1C, V3.1B
422503-021D	6010 Standard	V2.1D, V2.1E, V3.1C, V3.2A, V3.2B
422503-22A	6010 Ethernet	V2.1A
422503-22B	6010 Ethernet	V2.1B, V3.1A
422503-22C	6010 Ethernet	V2.1C, V3.1B
422503-22D	6010 Ethernet	V2.1D, V2.1E, V3.1C, V3.2A, V3.2B
422503-033B	6010 Standard	V3.3B
422503-034A	6010 Standard	V3.4A
422503-034B	6010 Standard	V3.4C
422503-034D	6010 Standard	V3.4D, V3.4E
422503-036A	6010 Standard	V3.6A
422503-037A	6010 Standard	V3.7A
422526-001A	7010	V3.1A
422526-001B	7010	V3.1B
422526-001C	7010	V3.1C, V3.2A, V3.2B
422526-033B	7010	V3.3B
422526-034A	7010	V3.4A
422526-034B	7010	V3.4C
422526-034D	7010	V3.4D, V3.4E
422526-036A	7010	V3.6A
422526-037A	7010	V3.7A
422544-033B	UDC	V3.3B
422544-034B	UDC	V3.4A
422547-033B	6010 Ethernet	V3.3B
422547-034A	6010 Ethernet	V3.4A
422547-034B	6010 Ethernet	V3.4C
422547-034D	6010 Ethernet	V3.4D, V3.4E
422547-036A	6010 Ethernet	V3.6A
422547-037A	6010 Ethernet	V3.7A
422547-038A	6010 Ethernet	V3.8A

1.4.3 Programming Executive Feature Support

The following table shows the versions of the Programming Executive in which the features or modules listed on the top are supported.

Table 1.3 - Features Supported by Versions of the AutoMax Programming Executive

Version of AutoMax Programming Executive	Feature																
	57C430A 57C431 Processors	57C435 Processor	Programmable Tick Rate	57C423 Common Memory	Works with Windows V3.1	Requires Windows V3.1	DC SD3000*** SF3000 Drives	AC SA500*** SA3000 Drives	UDC	UDC EM	SA3000 & SA3100 Const. HP & V/Hz	SB3000	VZ3000	A-B RIO Scanner	57C528 Remote I/O Head	AutoMax V2 OS Match
2.1-2.1A	yes*	no	no	no	no	no	no	no	no	no	no	no	no	no	no	n/a
2.1B	yes	no	no	no	no	no	no	no	no	no	no	no	no	no	no	n/a
2.1C	yes	no	no	no	no	no	no	no	no	no	no	no	no	no	no	n/a
2.1D	yes	yes	no	yes	no	no	no	no	no	no	no	no	no	no	no	n/a
2.1E	yes	yes	no	yes	no	no	no	no	no	no	no	no	no	no	no	n/a
2.1F	yes	yes	yes	yes	no	no	no	no	no	no	no	no	no	no	no	n/a
3.0A	yes	yes	no	no	no	no	no	no	no	no	no	no	no	no	no	2.1
3.1A	yes	yes	yes	no	no	no	no	no	no	no	no	no	no	no	no	2.1B
3.1B	yes	yes	yes	no	no	no	no	no	no	no	no	no	no	no	no	2.1C
3.1C	yes	yes	yes	yes	no	no	no	no	no	no	no	no	no	no	no	2.1D
3.2A	yes	yes	yes	yes	yes	no	yes	yes	yes	no	no	no	no	no	no	2.1D
3.3A-3.3F	yes	yes	yes	yes	yes	yes	yes	no	yes	no	no	no	no	no	no	n/a
3.4	yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	no	no	no	no	n/a
3.5	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes**	yes**	yes**	no	no	n/a
3.6	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes**	yes**	yes**	yes	no	n/a
3.7–3.8	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes**	yes**	yes**	yes	yes	n/a

* Works only with 57430-1K and earlier 57C430 Processors, and with 57431-C and earlier 57C431 Processors.

** With the release of V3.5, DPS drive software was uncoupled from the Executive software. This software must be purchased separately and installed on the personal computer along with V3.5 or later of the Executive.

***Must be purchased separately beginning with the release of V3.5.

1.4.4 ASD and the Programming Executive

Systems created with the AutoMax Software Designer (ASD) V8 and later can be converted into AutoMax Executive V3.x systems. Refer to section 5.1.8 for instructions. See instruction manual J-3615 for more information about ASD.

1.4.5 Converting from Previous Versions

Systems created with Version 2 of the AutoMax Programming Executive software can be converted into AutoMax Executive V3.x systems. Refer to section 5.3.8 for instructions. See instruction manual J-3684 for information about Version 2 Executive software.

Systems created with earlier versions of the AutoMax Version 3 Programming Executive software can be converted into AutoMax Executive V3.x systems. Refer to Appendix Q, Version Update Utility, for instructions.

1.4.6 Network Communication

Note that to fully use the network communication capability of the AutoMax Programming Executive software V3.x, all Network Communication modules on a given network must be M/N 57C404 (or later). An earlier version of the module (M/N 57404) can be used in the system, but the user will need to connect directly to the rack containing the Network module in order to communicate with that rack with a personal computer. M/N 57404 cannot be mixed with any other version on the same network. If Network module versions on a particular network are mixed, the communication capability of the network will be that of the earliest version of the Network module.

2.0 INTRODUCTION TO THE AutoMax DISTRIBUTED CONTROL SYSTEM

AutoMax is a programmable, microprocessor-based control system capable of performing real-time control with millisecond response time. Because AutoMax is modular, you can custom-configure the system to meet the specific requirements of your application. In addition, the Processor module operating system is included in the AutoMax Executive software and can be loaded from the same device used for writing application programs.

The available hardware modules include the following: computing, communications, digital input, digital output, analog input, analog output, motion control, and motor and drive control. In addition, the system can be configured to include many other commercially available Multibus-compatible modules.

Typically, an AutoMax system consists of a number of chassis, or racks, containing various modules. Up to 43 racks, each containing at least one Processor module, can be connected together as part of a control network using Network Communication modules. For drive control applications, each AutoMax rack can control up to 20 drives (10 UDC modules maximum) by using Universal Drive Controller (UDC) modules in a rack that also contains at least one AutoMax Processor module. Using Remote Communication modules, any rack containing at least one Processor module can control up to 7 remote I/O subsystems, which do not require Processor modules.

The system is designed to support the sharing of data among Processors in a single rack (up to four), as well as among Processors on different racks in a network. Applications can be distributed among multiple Processors and racks. Common data, such as I/O values, is accessible to all Processors simply by referencing the appropriate variable name in a task.

Application programs are created in an MS-DOS™ or MS-Windows environment using an 80386-compatible personal computer. AutoMax supports three different programming languages: Ladder Logic Language, Control Block Language, and Enhanced BASIC Language. Each of these languages is suited to a different type of task commonly found in the industrial and process control environment. AutoMax Control Block, BASIC, and PC tasks can be executed on an AutoMax Processor. Only UDC Control Block tasks can be executed on a Universal Drive Controller (UDC) module.

Ladder Logic Language, also known as PC (programmable control) Language, is used for sequential logic operations. Ladder Logic programming is very much like programming for a conventional programmable controller, incorporating standard ladder diagrams to establish the sequence and types of operations performed. Ladder Logic language is described in instruction manual J-3677.

Control Block language is used for programming control loops. The Control Block language consists of BASIC statements that contain special function calls, including amplifiers, integrators, function generators, PID controllers, etc. Control Block language is described in J-3676.

The Enhanced BASIC language is used for general programming, keyboard and CRT-based operator interfaces, and numeric processing. Enhanced BASIC language is described in J-3675.

Using the AutoMax Executive software, application programs, or tasks, created on the personal computer using Control Block or BASIC are then compiled and downloaded to one or more AutoMax Processor modules or UDC modules in a rack. Application tasks created using Ladder Logic do not need to be compiled before being downloaded. All application tasks can be run, monitored, modified, and stopped from the personal computer using the AutoMax Executive software.

See J-3675, J-3676 and J-3677 for more detailed information about programming for AutoMax systems.

2.1 ReSource AutoMax Programming Executive Overview

The remainder of this chapter describes the AutoMax Programming Executive software, which will be referred to as the Executive software from here on. See Appendix R for the **features** that are new in this release of the Executive software.

AutoMax Executive V3.x uses Microsoft Windows Version 3.1 (or later) to provide a graphic environment for all the software functions necessary to configure the hardware in your application and to create, organize, document, and troubleshoot application tasks on a personal computer. No special commands or syntax are used.

AutoMax Executive V3.x offline functions are contained within a set of four Windows applications: **System Configurator**, **Rack Configurator**, **Variable Configurator**, and **Task Manager**.

The **System Configurator** is used to organize the application, or system, into sections. Each section is a functional group of racks. For example, all of the racks for a physical section of a machine could be grouped into a section. See the heading “Application Organization” in this section for more information.

The **Rack Configurator** is used to configure the modules in a rack; the racks, heads, and rails connected to a remote I/O network; and the two-point modules in digital rails. The Rack Configurator uses a diagram of the AutoMax rack for configuring modules; a diagram of the remote I/O network for configuring remote racks, heads, and rails; and a diagram of the digital rail for configuring two-point modules. In AutoMax Executive V3.x, the Rack Configurator can be used to add the UDC module, specify the PMI and rail hardware, and enter drive parameters.

The **Variable Configurator** is used to map variables to I/O or memory points. It uses forms specific to the card on which the I/O or memory points reside. Mapping variables to physical locations on individual modules and to common memory locations allows the programmer to create application tasks referencing variable names instead of actual physical locations. This application essentially replaces the text configuration task used in earlier versions of the executive.

The **Task Manager** is used to develop application tasks. In the offline mode, the Task Manager is used to add, edit, compile, print, and verify tasks. In the online mode, it is used to load, save, run,

stop, and delete tasks on the AutoMax Processor or UDC module. Task error logs are viewed and cleared in the Task Manager.

A menu map for AutoMax Executive V3.x is shown in figure 2.1.

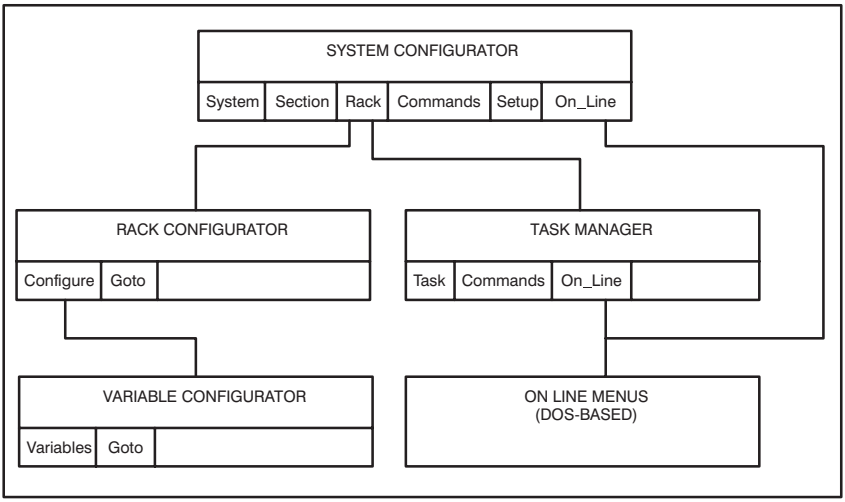


Figure 2.1 - AutoMax Menu Map

Application Organization

It is important to become familiar with the options available in each Windows application and with the hierarchy of information about your specific software application. You must first group all the software for a given application in a system. You then divide the system into sections. Sections are simply collections of racks (any number) and help organize the system in any manner appropriate to the application, e.g., related function or location.

The most basic level of application organization is the rack. The rack consists of the application tasks for the rack and all the information about the rack, including the modules and variables defined in it. Racks correspond directly to the AutoMax racks which are installed for the application.

To help you document the way in which your application is organized, documentation files can be created for a system, section, rack, or task. These text files can be used to explain design specifications, notes about troubleshooting, or any other information.

For users with numerous specific applications, AutoMax Executive V3.x also allows you to group systems into libraries. Libraries are used to store similar systems, e.g., plating systems. The default library is called AMXLIB and will be sufficient for most users.

The library/system/section/rack hierarchy allows you to structure each application so that it can be easily understood and its parts accessed quickly. It also allows you to carry out operations, e.g., copying, on an entire system, section, or rack.

2.2 Introduction to Configuring Racks and Variables

Once you create the system/section/rack structure for your application, you can begin to configure the hardware in your installation. Hardware configuration consists of adding modules to each rack (including network and remote racks, UDC modules, and PMI rail hardware) to reflect the actual installation. AutoMax Executive V3.x checks that modules are added properly. For example, a second AutoMax Processor module can be added only after a Common Memory Module (M/N 57C413 or 57C423) has been added. If your application includes modules that are not supported in V3.x, you can use “generic” modules. There are three generic modules: one with 32,768 registers (GEN32K), one with 8000H registers in hex (GEN32KH), and one with 32 registers (GEN32) for use in a remote I/O rack (or in a main rack when 32 registers are sufficient). Note that GEN32K and GEN32KH modules can be used only in a rack that contains an AutoMax Processor module.

After you have added a module, you can map variables to I/O points or common memory on that module using “forms” tailored as required for the specific register organization on each module. This variable configuration “map,” along with information about the application tasks for the rack, is stored in databases, and must be compiled and downloaded to the rack along with application tasks before application tasks can run in the rack. This configuration information is sometimes called the configuration task.

After you have completed the physical configuration of your system, you can generate a bill of material which lists all of the racks, heads, rails, and modules used in the system along with the needed batteries, cables, and optional hardware and software. See Appendix L for additional information.

2.3 Introduction to Creating Application Tasks

After you have configured I/O points and common memory using variable names, you can create application tasks using these variable names instead of locations. You can create BASIC tasks, PC/Ladder Logic tasks, and Control Block tasks. BASIC and Control Block tasks are stored in text files which are edited using a text editor; PC/Ladder Logic tasks are stored in binary files and edited using a custom editor. A text editor (Norton Editor) and the PC Editor are both included with the AutoMax Programming Executive software. In addition, you can create “include” files which are added to a BASIC or Block task when it is compiled (by using the BASIC language INCLUDE statement).

Once the variables have been configured and the application tasks created, you can compile the tasks to create object code that will run on the AutoMax Processor or UDC module(s) in the rack. PC/Ladder Logic tasks are stored as object code and do not need to be compiled.

Drive Control

Tasks that control Distributed Power Drives (DPS) are stored and executed on UDC (Universal Drive Controller) modules in the rack. UDC tasks are written in control block language. They are referred to as "UDC tasks." See the appropriate DPS programming and configuration instruction manual for more information about UDC tasks.

Multi-Tasking

AutoMax Processors are capable of multi-tasking. Each AutoMax Processor can execute multiple application tasks based on a priority basis, sharing common data between tasks. Application task execution can also be coordinated using BASIC language statements and functions in both BASIC and Control Block tasks.

Each UDC module can execute two independent Control Block tasks to control two separate drives of any type, usually referred to as Drive A and Drive B. UDC tasks A and B execute sequentially (task A executes, then B) based on a user-defined interval of 500 μsec CCLK signal. If only one task is required, i.e., the UDC module will control only one drive, the drive can be connected through either channel A or B.

The UDC tasks on one module share the data in the dual port memory. UDC tasks cannot, however, share data with tasks on other UDC modules, or with tasks on AutoMax Processors, unless an AutoMax task reads from or writes to the UDC's dual port memory. Only AutoMax tasks can move data between two or more UDC tasks on different UDCs, and between UDC tasks and AutoMax tasks.

The following table illustrates the differences between UDC modules and AutoMax Processors.

UDC Module and AutoMax Processor Comparison

	Maximum # in a Rack	**Maximum # of Tasks****	**Programming Languages**
AutoMax Processor	4	32	BASIC, Control Block* or PC/Ladder Logic
UDC Module	10	2 (1 each for drives A/B)	Control Block*

* A subset of BASIC statements and functions is permitted. The two types of Control Block tasks (UDC and AutoMax) support slightly different subsets. These are listed in the Control Block instruction manual (J-3676) and in Appendix C of this manual.)

** Each rack can have a maximum of 32 tasks loaded into it. This total does not include BASIC "include" tasks.

2.4 Introduction to Online Operations

Before you can go online to any rack in the system, the operating system, or runbase, for the AutoMax Processor and UDC module(s) must be loaded to the rack. In general, you will only be concerned with the file types listed below for purposes of loading to the rack.

1. Operating system (OS) file
   Each Processor module and UDC module must have an operating system file. The operating system for each Processor module in a rack is usually the same. The Processor operating system is included in the Executive software. UDC OS files are included in the separately-purchased DPS software options. Note that UDC operating systems contain within them operating systems for the Power Module Interface (PMI) hardware they are connected to. See section 5.4.3 for more information.

2. Configuration file
   One configuration file is required for each rack. This file is stored on the Processor module. If there is more than one Processor module in the rack, the file is stored on the Common Memory module. Configuration files are identified by the extension ".CNF" in the file name. After they have been prepared for loading to the rack, the configuration files will have extension .OBJ. See section 5.4.3 for more information.

3. Application task files
   Application tasks are stored and executed on Processor modules and Universal Drive Controller modules in the rack. Each Processor module may store/execute more than one task. Each Universal Drive Controller module may store/execute one or two tasks. Application tasks are identified by the one of the following extensions in the file name: .BLK , .PC, or .BAS. After the files have been prepared for loading to the rack, however, they will have extensions .OBJ, .PC, and .OBJ, respectively. See chapter 8 for more information.

4. Parameter object file
   Each Universal Drive Controller module must have one parameter object file. Parameter object files are identified by the extension ".POB" in the file name. See chapter 8 for more information.

2.5 AutoMax Help Screens

The AutoMax offline "help" feature uses the same menu structure and options as Windows Help. For a description of the Windows Help feature, refer to the *Microsoft Windows User's Guide*.

Help is available for each of the four applications in AutoMax Executive V3.x (System Configurator, Rack Configurator, Variable Configurator, and Task Manager). Help is also available for most dialog boxes. In addition, pop-up help is available for key terms. Simply press F1 (or click on Help) from any screen or dialog box listing that option. F1 is reserved exclusively for accessing the Help feature. Note that help for online commands, also accessed using F1, consists of text files and does not use the Windows Help menu structure.

3.0 NOTES ON INSTALLATION

This section describes what you need to know before installing the Programming Executive software. Refer to instruction manual J2-3107 for the installation procedure.

WARNING

QUALIFIED PERSONNEL MUST READ AND UNDERSTAND THE APPLICABLE AutoMax AND DCS 5000 INSTRUCTION MANUALS IN THEIR ENTIRETY PRIOR TO INSTALLING, ADJUSTING, OPERATING, AND SERVICING THE AutoMax SYSTEM AND THE MACHINERY THAT IT OPERATES. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY OR DAMAGE TO EQUIPMENT.

Requirements for installing the Programming Executive:

1. Your computer must be an IBM-compatible 80386-based (or higher) system with at least 2MB RAM (4MB recommended) and a 40MB hard drive.
2. A serial port must be available for communication with the AutoMax Processor. If you want to use a serial mouse while your computer is connected to the Processor, you will need a second serial port.
3. The operating system on your computer must be version 5.0 or later of MS-DOS (or equivalent). IBM OS/2™ is not supported.
4. Your computer must have V3.1 or later of Microsoft Windows software or Windows 95. Windows must be running in enhanced mode.

3.1 Installing the Windows Software on the Personal Computer

Windows 3.1, 3.11, or Windows 95 must be installed on your personal computer before you can install AutoMax Executive V3.x. Read this section carefully before beginning to install the Windows software. If you need additional information, refer to the *Microsoft Windows User's Guide*. **If Windows is already loaded on your personal computer, go on to section 3.2.**

Disk 1 of the Microsoft Windows disk set contains a program called Setup, which guides you through the installation. It will evaluate your computer system to determine what kind of equipment you have. It will then copy essential Windows files onto your hard disk.

After the preliminary installation, Setup will start Windows automatically, installing additional files, and asking you to verify and provide information about the following:

- The directory where you want to store Windows
- The type of computer you are using
- Your monitor
- Your mouse, if any
- Your keyboard and keyboard layout

- Your language (English, Spanish, etc. Some Windows applications use this setting for language-specific tasks, such as sorting or case conversion.)
- The kind of local area network you are on, if any
- The printer and printer port you use
- Applications on your hard disk that you want to run with Windows

The following procedure can be used to install the Windows software.

Step 1. Insert the diskette labeled “Windows Disk 1” in drive A:.

Step 2. Type **A:SETUP**<CR>

Step 3. Follow the directions on the screen. If you have any questions about any of the procedures or options, you can press F1 to access an online help feature.

3.2 Installing and Running the Programming Executive Software on the Personal Computer

Before you load the Programming Executive software onto the personal computer, read carefully both the instruction manual describing the operation of the personal computer you will be using and the manual describing the DOS operating system. Note that you cannot install the Programming Executive software if Windows V3.1, 3.11, or Windows 95 is not already installed on your personal computer.

You will need to refer to the installation instructions in J2-3107, found in binder J2-3108, for the specific steps required to install the software. Read the instructions carefully and follow all the steps in the order given to install the software properly.

For Personal Computers with other AutoMax Executives Installed

Note that you can install the AutoMax Executive V3.x on the same personal computer with any other version of the Executive software. The following guidelines apply:

Running Two Versions of V3.5 or Later
Change the EXEC_DRIVE entry in the AUTOMAX.INI file each time you want to switch between the two versions.

Running Two Versions, V3.0A to V3.4E
Close windows and change the DCSEXDRV environment variable in the AUTOEXEC.BAT file. Reboot your computer. Then restart Windows.

Running Two Versions, V3.5 or Later and V3.0A - V3.4E
No special commands are required. Simply click the desired icon.

Running Two Versions, V3.X and V2.X
No special commands are required. Simply click the desired icon to run V3. Type AUTOMAX2 at the DOS prompt to run V2.

Running the Programming Executive
Once you have installed the Executive, you can run it by clicking the AMX V3 icon in the AutoMax Programmer or typing “AUTOMAX3” at the DOS prompt. Typing “AUTOMAX3” will also invoke Windows.

To run AutoMax V3 from Windows 95, click on Start and choose:
Programs > Automax V3.x > AutoMax V3

To set up AutoMax V3 if V4 is also installed, follow these steps:

Step 1. Copy the file AMX3RUN.EXE from the V4 installation directory (typically C:\RPE\AMX4) to the V3 installation directory (typically C:\RPE\AMX3).

Step 2. Navigate to the AutoMax V3.x program folder and select the AutoMax V3 icon.

Step 3. From the File menu choose Properties. The AutoMax V3 Properties dialog is displayed.

Step 4. Select the Shortcut tab and type the following in the Target field:
C:\RPE\AMX3\AMX3RUN.EXE II.EXE

Step 5. Click on the Change Icon button and type the following in the File Name field:
C:\RPE\AMX3\II.EXE
Click OK.

Repeat this procedure as needed for the other icons in the AutoMax V3.x program folder. AutoMax V3 can now be run from the Start menu as described.

When you run AutoMax Executive V3.x, the first screen displayed after the name and copyright information is the System Configurator. Choose the Setup menu and then select AutoMax to run the AutoMax configuration procedure. This will allow you to make changes to the AutoMax Executive configuration. See section 5.5.2 for detailed instructions.

3.3 Installing a Mouse

In addition to the AutoMax Executive and Windows software, it is recommended that you also install a mouse and the software required to support it. Although a mouse is not required, the Executive software is easier to use with a mouse. A list of compatible mice is included with the Windows software package.

A mouse is used to move the cursor or pointer on your screen to make selections or specify areas for data entry. There are two different types of mice, a bus mouse and a serial mouse. Each type has its own installation procedure.

A bus mouse requires a bus card that must be configured, usually by setting jumpers, before it is installed in an expansion slot of your computer. Each jumper is placed over a set of pins to make an electrical connection. Refer to the instructions that are provided with your mouse for the appropriate jumper settings for your computer. The mouse cable then plugs into a port on the back of the bus card, visible through the back of the computer.

A serial mouse is installed by plugging the mouse into a serial port on the back of your computer (COM1 or COM2). Some computers have a 9-pin serial port, others a 25-pin port. If the serial port and mouse connector do not have the same number of pins, an adapter can be used to connect the mouse cable connector to the serial port.

For both kinds of mice, a software driver is necessary to enable the mouse and the Executive software to communicate. The mouse

usually comes with a diskette containing the required files, including the driver program (typically MOUSE.COM). These files must be loaded onto your hard drive and called from your AUTOEXEC.BAT file when you boot up your computer.

Refer to the documentation provided with your mouse for the proper installation procedure.

3.4 Connecting the AutoMax Processor to the Personal Computer

For all online operations, such as running or monitoring application tasks and loading operating systems, the personal computer must be connected with a serial cable to an AutoMax Processor module (M/N 57C430A, 57C431, or 57C435) in the rack or via the AutoMax Network using an AutoMax PC Link Interface module. Section 3.4.1 describes serial connection; section 3.4.2 describes the PC Link module.

3.4.1 Serial Connection

The cable connection is usually made between the COM1 port on the personal computer and the upper port, labeled "Programmer/Port B", on the Processor module in the rack. If you want to access the Processor through COM2 or via modem, you can select the "Com 2" parameter on the Communications Setup screen. See 5.5.3 for more information. See Appendix F in the AutoMax Processor module instruction manual (J-3650) for information on using a modem with an AutoMax system. If there are multiple AutoMax Processor modules in the rack, you must connect your personal computer to the upper port of the leftmost Processor module. Through the single connection at the leftmost Processor, the personal computer can communicate with any Processor in the rack, with all I/O modules located in this rack, and with all remote I/O modules connected through a Remote Communication module (M/N 57C416) to this rack. Refer to Appendix A in this instruction manual for a description of the pins on the Programmer/Port B port when it is used to communicate with the personal computer.

Baud Rate

The default baud rate for communication between the personal computer and the Processor is 9600 (19200 for M/N 57435 Processor). The operating system for the Processor module(s), as well as application tasks that are loaded using the /ALL option, will be loaded at 9600 baud if there are no tasks currently running in the rack.

You can change the baud rate by editing the Baud Rate parameter on the Communication Setup screen. See section 5.5.3 for more information. The allowable baud rates are 1200, 2400, 4800, 9600 and 19200 baud. You can override the baud rate settings at any time through the BAUD RATE option available when you are online with the Processor(s) (see 12.1).

Cable Connection

You will need to purchase cable M/N 61C127 or manufacture a serial cable with a 9-pin connector on one end for the personal computer and a 25-pin connector on the other end for the Processor module. Use the cable to connect the serial port on the personal computer to

the upper connection, labeled “PROGRAMMER/PORT B”, on the leftmost Processor module in the rack. See Appendix A for a description of the pins and directions for making your own cable.

3.4.2 Using a PC Link Module

Version 3.3C (and later) of the AutoMax Programming Executive software supports the use of the AutoMax PC Link Interface module (M/N 57C445) to communicate with a rack over the AutoMax Network. The PC Link module is functionally equivalent to a Network Communications module, and allows the personal computer in which it is installed to function as a drop on the AutoMax Network. By using AutoMax PC Link modules, up to four PC workstations can communicate with a single AutoMax rack.

Use the procedures described in the AutoMax PC Link Interface Module instruction manual (J2-3011) to configure and install the PC Link module in the personal computer, and to load the PC Link software onto the personal computer’s hard disk. **Note that the PC Link software (included with the PC Link module) must be loaded to the same directory as the AutoMax Programming Executive software.** This will ensure that the PC Link module is recognized by the Programming Executive.

Section 5.5.3 of this manual describes how to configure the Programming Executive to use the PC Link module for online communication.

3.5 User Ports

All AutoMax Processor ports in the rack, other than the port used to connect the Processor to the personal computer, are available for use by the application programs running on the respective AutoMax Processor modules. See 10.6 for more information on user ports.

The ports are accessed using the OPEN statement (OPEN “PORTA” or OPEN “PORTB”) in BASIC tasks. Refer to J-3675, the Enhanced BASIC Language Instruction Manual, for more information on the OPEN statement. Refer to the AutoMax Processor module data sheet (J-3729), which accompanied the Processor module, or the AutoMax Processor Instruction Manual (J-3650) for a description of the pins on user ports.

4.0 GETTING AROUND IN THE AutoMax EXECUTIVE AND WINDOWS

All operations in Windows can be executed with either the keyboard or the mouse. The sections that follow describe techniques used to make selections and execute commands. All techniques will be described using both the keyboard and mouse. The *Microsoft Windows User's Guide* provides additional information on basic skills and should be referred to for any specific questions.

Keyboard Keys

In this manual, keynames are capitalized (ESC, ENTER, CTRL). A plus sign (+) used between two keynames indicates that those keys must be pressed at the same time. For example, "Press ALT+ESC" means you should press the ALT key and hold it down while you press the ESC key and release it. Then release the ALT key. A comma between two keys indicates that those keys must be pressed sequentially. For example, "Press ALT,SPACEBAR" means you should press the ALT key and release it; then press the SPACEBAR and release it.

The DIRECTION keys are the four arrows on the keyboard that are used to move the selection, pointer, or cursor on the screen. The keys are written as UP, DOWN, LEFT and RIGHT. When using the keyboard, you usually use the DIRECTION keys to move the cursor. ENTER is usually used to execute a chosen command.

Mouse Buttons

When using a mouse, you use the left mouse button for most functions. (It is possible to swap the left and right button functions. See the *Microsoft Windows User's Guide* for instructions on using the Windows Control Panel.) To point, move the mouse until the pointer rests on what you want. To click, press the button and release it. To click-and-drag, press the button and hold it down while moving the mouse. To double-click, press and release the button twice in rapid succession.

4.1 Making Selections

A basic concept in the Programming Executive is that you must first select an item before you can perform an operation. For example, you select a window and then work in it, or you select an item, e.g., an application task, and then do something to it, e.g., delete it.

4.2 Choosing Commands from Menus

Windows commands are organized into menus. Menu names are listed on the menu bar which runs across the top of the window under the window title bar (see figure 4.1). The title bar will usually list the name of the application; the currently selected system, section, and rack; and the library drive and directory. After selecting a menu, you can choose a command from that menu. A description of the highlighted command will then be displayed on the title bar. Choosing a command executes the command.

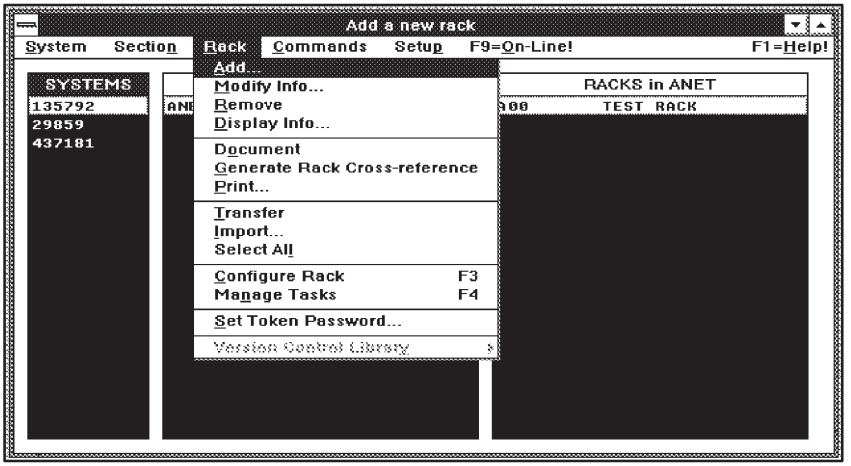


Figure 4.1 - Choosing a Command

If you are using the keyboard, there are two ways to choose commands from Windows menus: the basic method and the direct access method. The basic method uses the DIRECTION keys. To choose a command, follow these steps:

Step 1. Press ALT,LEFT (or RIGHT) to select the menu on the menu bar.

Step 2. Press ENTER to display the menu.

Step 3. Use UP or DOWN to select the command you want.

Step 4. Press ENTER to execute the command.

The direct access method can be used when there is an underlined letter in the menu or command name. Follow these steps:

Step 1. Press ALT+(the underlined letter) in the menu name.

Step 2. Press the underlined letter in the command name.

Some commands have a 'shortcut' key combination listed next to the command name (e.g. Configure Rack F3, Add ^A). These commands can be executed directly (without first selecting the menu) by entering the shortcut key combination. For example, the Rack Configurator can be accessed from the System Configurator by pressing F3; Add (a Module to a rack) can be accessed from the Rack Configurator by pressing CTRL+A.

If you are using the mouse, choose a command as follows:

Step 1. Click the menu name on the menu bar.

Step 2. Click the command name.

4.3 Canceling a Menu

To use the keyboard to cancel a menu, press ESC.

If you are using a mouse, cancel by clicking a blank area of the window.

4.4 The Control Menu

The Control menu is used to move and re-size windows on the screen, close applications, and access the Task List. This menu is symbolized by the box in the upper-left corner of each window. To select the Control menu from the keyboard, press ALT,SPACEBAR. You can then use either the standard method or direct access method to choose a command. If you are using a mouse, click the Control menu box in the upper-left corner of the window, then click the command you want to choose, as you would for any other menu.

The Control menu commands for AutoMax Executive V3.x are as follows:

- **Restore** restores the window to its former size after it has been enlarged (using the Maximize command) or shrunk to an icon (using the Minimize command).
- **Move** enables you to use the keyboard to move the window to another position on the desktop (the screen background for Windows on which windows, icons, and dialog boxes appear).
- **Size** enables you to use the keyboard to change the size of the window.
- **Minimize** shrinks the window to an icon.
- **Maximize** enlarges the window to its maximum size.
- **Close** closes the window.
- **Switch To...** accesses the Task List dialog box, which enables you to change applications, or to rearrange windows and icons on your desktop.

  To close an application using the mouse, simply double-click the Control menu box. See figure 4.2.

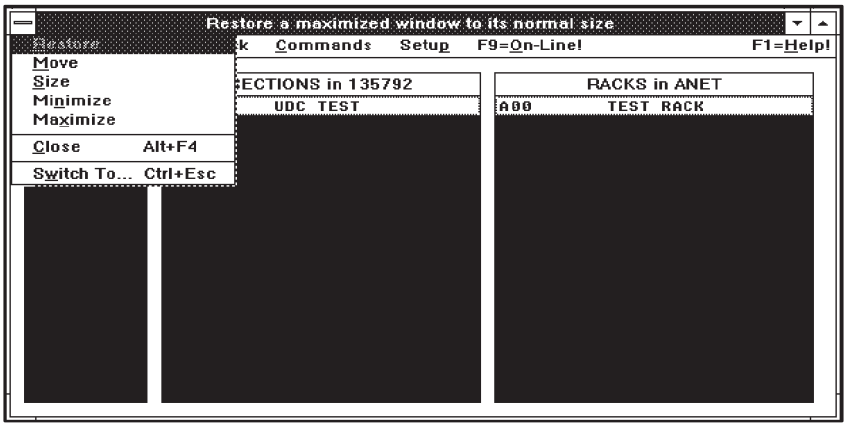


Figure 4.2 - The Control Menu

4.5 Inactive Commands

If a command name is displayed in a lighter text on the menu, it means that the command is currently not accessible. You may have to select something before you can use the command. See figure 4.3.

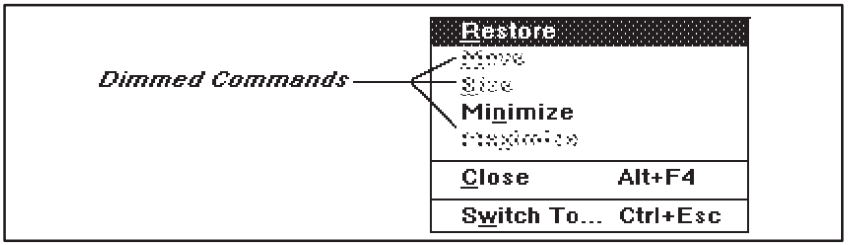


Figure 4.3 - Dimmed Commands

4.6 Making Multiple Selections from Lists

When you work with AutoMax Executive V3.x, you must select the item that the next command or action will affect. Items are usually selected one at a time, but multiple items can be selected and acted upon by some commands. Each item will be highlighted as it is selected. Note that the items selected do not have to be contiguous; you can select any number of items that appear on the same screen display.

If you are using the keyboard, use the DIRECTION keys to locate each item you want to select. To select multiple items:

Step 1. Use CTRL+DOWN (or UP) key to locate the first item you want to select. A box comprised of dotted lines indicates the item to be selected.

Step 2. Press SPACEBAR to select the item. The item will be highlighted.

Step 3. Repeat the preceding steps for each additional item to be selected.

To cancel multiple selections, press UP or DOWN without pressing CTRL. To un-select a previously selected item, locate the item using UP or DOWN, then press CTRL+SPACEBAR.

If you are using a mouse, click on the first item you want to select. To select multiple items, press and hold CTRL while clicking on each subsequent item. Items will be highlighted as you click on them. To unselect a previously selected item, press and hold CTRL while clicking on the item. To cancel multiple selections, click on any item without pressing CTRL.

4.7 Dialog Boxes

A **dialog box** is displayed by AutoMax when additional information is needed to execute a command. The dialog box contains areas where you can enter the needed information. The areas displayed will vary depending on the types of information needed. A dialog box can contain text boxes, list boxes, check boxes, option buttons, and command buttons. See figure 4.4.

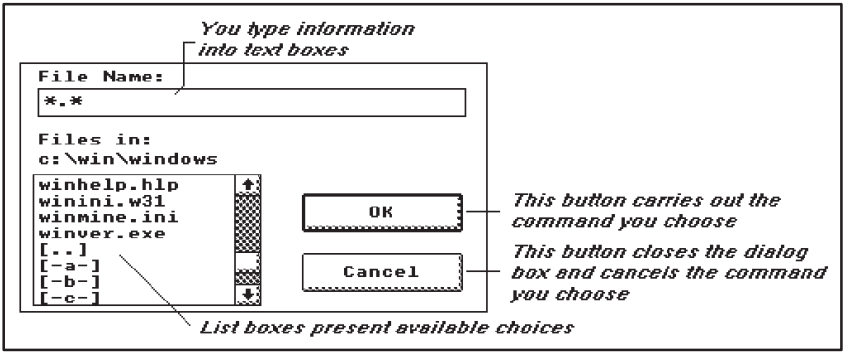


Figure 4.4 - Dialog Boxes, Part 1

A **text box** is used to enter text information. The text appears to the left of a flashing vertical line which pushes the text to the right as you type. The DIRECTION keys can be used to move the vertical line within the text to edit it. If the box already contains text when you move to it, and the text is selected, any text you type will replace it.

A **list box** contains the names of available choices. If file names are included, they will be listed with their extensions. Drives and subdirectories are indicated by brackets; hyphens denote a drive. For example, [-A-] represents drive A; [PIF] represents subdirectory PIF. The parent directory is represented by two periods within brackets ([..]). The list box may have scroll bars (up and down arrows) if all the choices don't fit in the box.

Command buttons carry out commands when chosen; they are labeled (OK, Cancel, Open, Reset, etc.) to indicate what the buttons do. See figure 4.5.

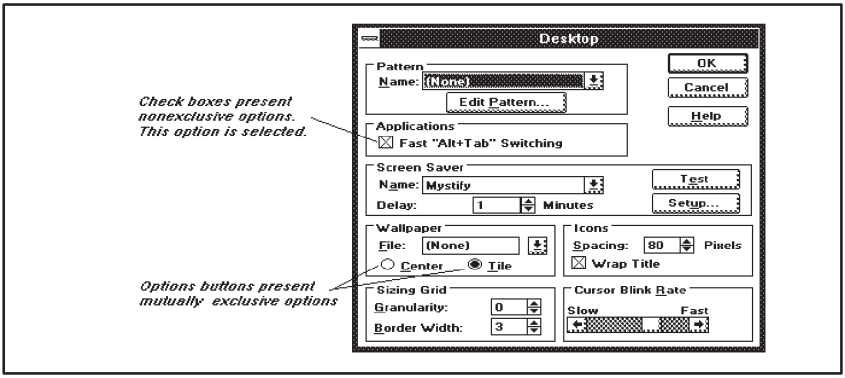


Figure 4.5 - Dialog Boxes, Part 2

Option buttons let you select options for a command. You can select only one option in a group of option buttons.

Check boxes allow you to choose multiple options for a command. Each check box is a separate area in the dialog box.

When moving from one area of a dialog box to another, you will notice that the area selected is usually marked by a box enclosed by dotted lines. You can press TAB to move from area to area. SHIFT+TAB moves to areas in reverse order. To select an item within a group in the dialog box, use the DIRECTION keys. In a list box, use the DIRECTION keys to move up or down in the list. The item you move to will be highlighted to show it is selected. If you are using a mouse, you can select any area in a dialog box by clicking the desired area. If there are several items in a group, the item you click will be selected.

You can also use the DIRECTION keys to move through a group of option buttons. Press SPACEBAR to select the highlighted item. If you are using a mouse, click the item you want to select. If the item is not visible, use the scroll bar to bring it into view. To select an option, simply click it.

4.8 Moving a Window Or Icon

You can move windows or icons to a different location on the "desktop," i.e., the screen background for Windows.

If you are using the keyboard:

Step 1. Press ALT+ESC to select the window or icon you want to move.

Step 2. Press ALT+SPACEBAR to access the Control menu.

Step 3. Press M (Move).

Step 4. Use the DIRECTION keys to move the window or icon.

Step 5. Press ENTER to complete the move.

If you are using a mouse, drag the title bar of the window or drag the icon.

4.9 Changing the Size of a Window

You can change the size and shape of the windows on the desktop.

If you are using the keyboard:

Step 1. Press ALT+ESC to select the window you want to re-size.

Step 2. Press ALT+SPACEBAR to access the Control menu.

Step 3. Press S (Size).

Step 4. Use the DIRECTION keys to re-size the window.

Step 5. Press ENTER when the window is the size you want.

If you are using a mouse, drag the corner or border of the window until it is the size you want.

4.10 Enlarging a Window

You can enlarge a window to fill the entire desktop.

If you are using the keyboard:

Step 1. Press ALT+ESC to select the window you want to enlarge.

Step 2. Press ALT+SPACEBAR to access the Control menu.

Step 3. Press X (Maximize).

If you are using a mouse, click the Maximize box (up pointing arrow) in the upper-right corner of the window. See figure 4.6.

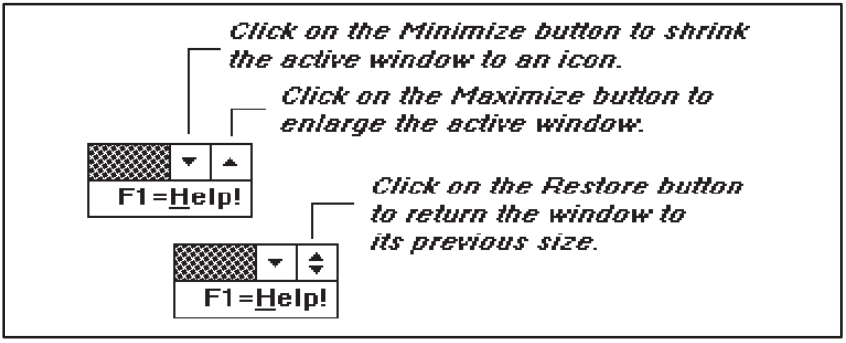


Figure 4.6 - Maximize, Minimize, and Restore Buttons

4.11 Shrinking a Window to an Icon

You can shrink a window, i.e., an application, to an icon when you are finished working with it but want it available for later use.

If you are using the keyboard:

Step 1. Press ALT+ESC to select the window you want to shrink.

Step 2. Press ALT+SPACEBAR to access the Control menu.

Step 3. Press N (Minimize).

If you are using a mouse, click the Minimize box (down pointing arrow) in the upper-right corner of the window.

4.12 Restoring a Window or Icon

You can return an window or icon to the size it was before you enlarged it or shrank it to an icon.

If you are using the keyboard to restore a window:

Step 1. Press ALT+SPACEBAR to access the Control menu.

Step 2. Press R (Restore).

To restore an icon:

Step 1. Press ALT+ESC to select the icon you want to restore.

Step 2. Press ALT+SPACEBAR to access the Control menu.

Step 3. Press R (Restore).

If you are using a mouse, click the Restore box (double-headed arrow) in the upper-right corner to restore a window. Double click an icon to restore it.

4.13 Using Scroll Bars

Application windows and list boxes may have scroll bars if all the information won't fit in the window.

If you are using the keyboard, press the DIRECTION key in the direction you want to scroll. You can also use the following keys:

Up one screen, press PAGE UP. Down one screen, press PAGE DOWN. Top of a list, press HOME. Bottom of a list, press END.

If you are using a mouse, drag the scroll box in the direction you want to scroll. You can scroll one line at a time by clicking the scroll arrows at either end of the scroll bar. You can scroll one screen at a time by clicking the scroll bar between the scroll box and a scroll arrow. See figure 4.7.

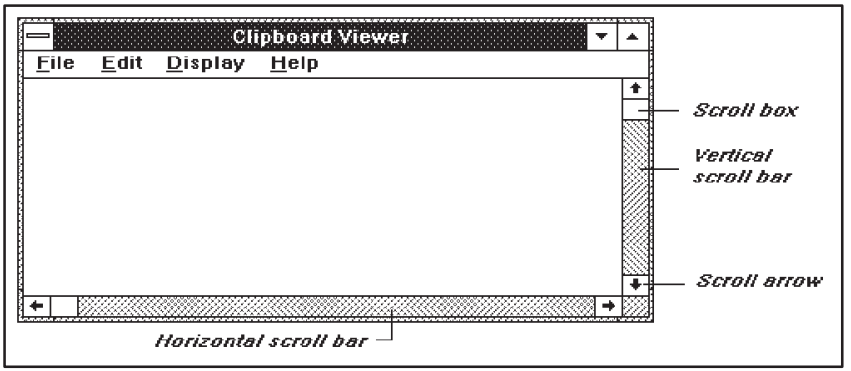


Figure 4.7 - Scroll Bars

4.14 Closing an Application

You can use the Exit command to close a Windows application. Any changes that have been made in the application will be saved, and you will return to the application you were running prior to the present application. Closing the System Configurator application will return you to DOS unless you have elected, during AutoMax setup, to exit to the Windows Program Manager group. See section 5.5.2 for more information. See figure 4.8.

If you are using the keyboard:

Step 1. Press ALT+(underlined letter) of the left-most menu on the screen.

Step 2. Press E to execute the Exit command.

If you are using a mouse, double click the Control menu box.

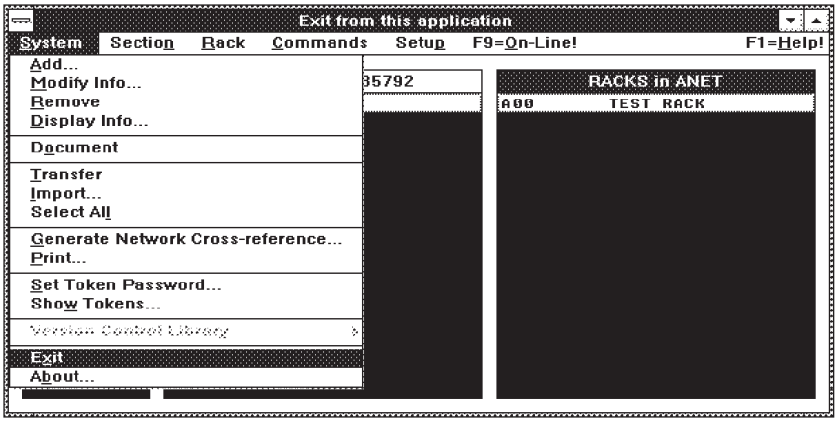


Figure 4.8 - Closing an Application

4.15 Ending a Windows Session

You can exit Windows from the Program Manager. When you exit a Windows session, you return to DOS. See figure 4.9.

If you are using the keyboard:

Step 1. Press ALT,F to select the File menu.

Step 2. Press X to select the Exit Windows command. A dialog box asks you to confirm that you want to end the session.

Step 3. Choose the OK button to end your session.

If you are using a mouse:

Step 1. Double click the Control menu box. A dialog box asks you to confirm that you want to end the session.

Step 2. Click the OK button to end your session.

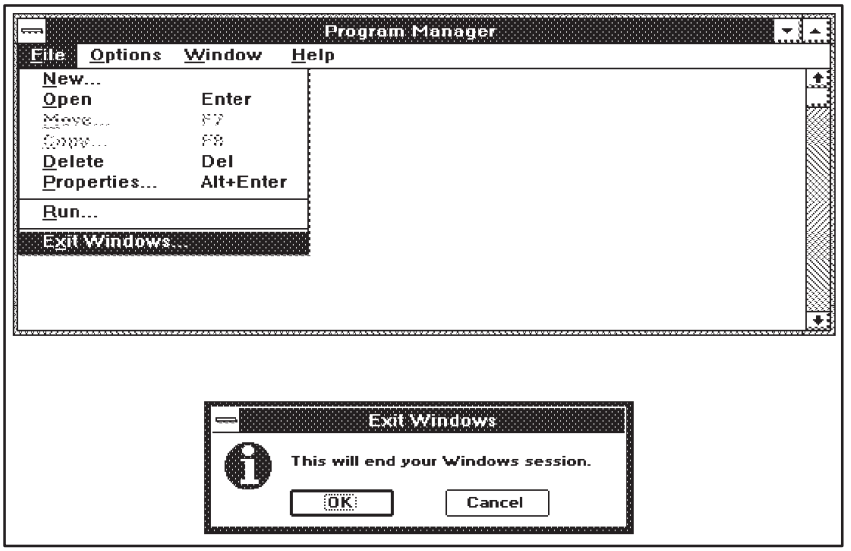


Figure 4.9 - Ending a Windows Session

4.16 Using the Mouse Accelerator (Double-Clicking)

The mouse can be used to move quickly through the various levels of the rack and variable configuration process. Double-clicking the left button on the desired object accesses the next logical lower level of the configuration process.

For example, beginning from the master rack in the Rack Configurator, double-clicking the left mouse button on a Remote I/O module will access the Remote I/O Network diagram. Double-clicking a remote I/O rack will display the remote rack. You can then add modules to the remote rack. Double-clicking a configurable module will access the Variable Configurator. See figure 4.10.

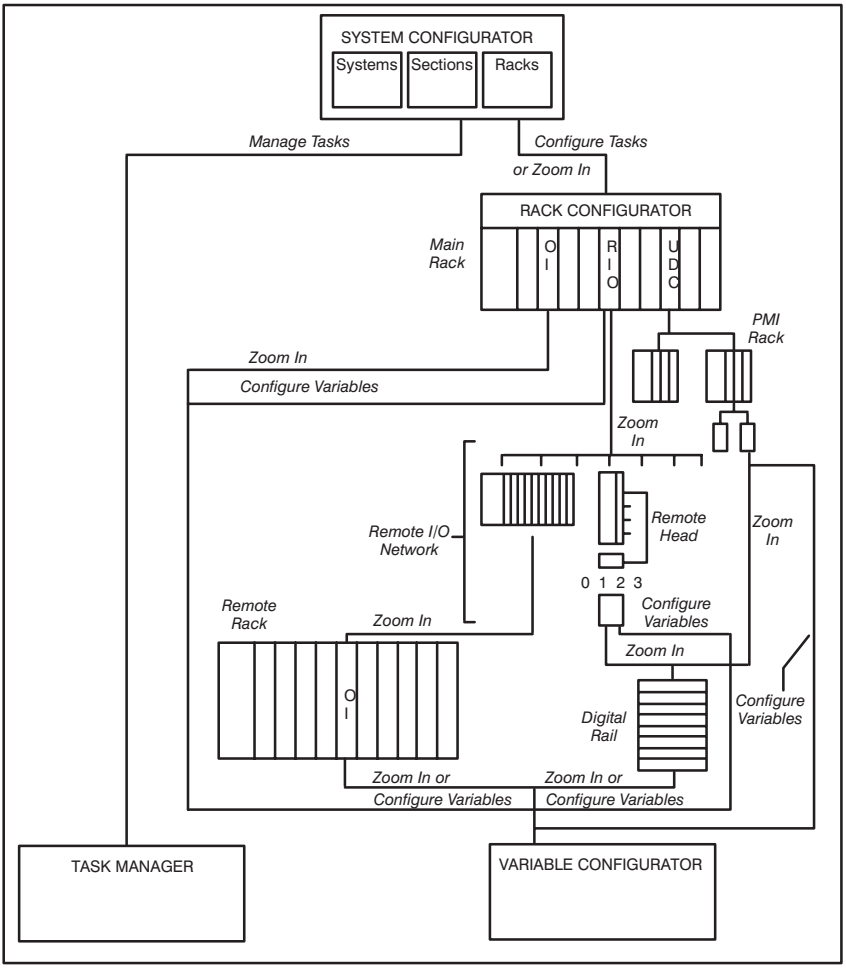


Figure 4.10 - Use of the Mouse Accelerator

Double-clicking the right button returns you to the next higher level in the configuration process.

4.17 Using the DOS Command Prompt

You must use the Windows Program Manager to access the DOS prompt. The DOS prompt appears as a program item in the Program Manager Main Group. To start the DOS prompt:

Step 1. Access the Control menu.

Step 2. Select **Switch To**. The Task List dialog box will be displayed.

Step 3. Select Program Manager. The Program Manager will be displayed.

Step 4. Select the DOS Prompt from the Main Group. The DOS prompt will be displayed.

To return to the Executive:

Step 1. Type **Exit** at the DOS prompt. The Program Manager will be displayed.

Step 2. Access the Control menu.

Step 3. Select **Switch To** and then select the AutoMax Executive software application you were running before you exited to DOS.

4.18 Directory Structure

The library is the base directory under which all the AutoMax systems are stored. The default drive and library name are specified as part of the Setup procedure for the AutoMax Executive software. If you want to create a new library or change the default (selected) library or drive, you must use Setup. See section 5.5.2 for more information.

The system database files are stored as follows:

d:\library\system

where:

d	is the disk drive
library	is the name of the library
system	is the name of the system

The system database files define the system and the sections, racks, and network-wide variables in the system. The \system subdirectory includes the documentation files for the system and for the sections and racks within the system.

The rack database files are stored in d:\library\system\rack, where “rack” is the name of the rack. The rack database files define the modules, tasks, and variables in the rack. The \rack subdirectory includes the source code, object code, cross reference files, and documentation files for the application tasks in the rack.

Backup of the AutoMax Executive software library directory and all its subdirectories should be done on a regular basis using the DOS BACKUP command.

4.19 Database Files

The AutoMax Executive software stores the data for AutoMax systems in dBASE III-compatible database files. There are four databases for each system and five databases for each rack in a given system. Database files are designated by a dollar sign ($) as the first character of the filename and a .DBF extension. Index files (which contain pointers to where in the database information is stored) also have a dollar sign ($) as the first character of the filename, but they are assigned a .NDX extension.

For each system:

$SYSTEM.DBF $SYSTEM1.NDX	defines the system
$SECT.DBF $SECT1.NDX	defines the sections in the system
$RACK.DBF $RACK1.NDX $RACK2.NDX	defines the racks in the system
$NET.DBF $NET1.NDX $NET2.NDX $NET3.NDX	defines the network-wide variables in the system

For each rack in a system:

$CARD.DBF $CARD1.NDX	defines the modules in the master rack and any remote racks
$IORACK.DBF $IORACK1.NDX	defines any remote racks connected to the rack
$TASKS.DBF $TASKS1.NDX	defines the tasks in the rack
$SWVAR.DBF $SWVAR1.NDX $SWVAR2.NDX $SWVAR3.NDX	defines the variables configured in the rack
$PARAM.DBF $PARAM1.NDX	defines parameters for UDC module

See Appendix F for a description of the relevant fields for all of the AutoMax Executive databases.

4.20 AutoMax File Types

In addition to the files described in section 4.18, the following file types may be created when you use the Programming Executive. The file type is determined by the file extension. AutoMax uses the file extension to determine the particular editor (text editor or PC editor) to be used. When you create a file, the Executive appends one of the following file extensions to designate the file type:

.BAS	BASIC tasks
.BLK	Control Block tasks
.CNF	Configuration files
.PC	PC/Ladder Logic tasks
.INC	Include files (see BASIC statement INLCUDE)
.SDC	System documentation files
.CDC	Section documentation files
.RDC	Rack documentation files
.TDC	Task documentation files
.IDC	Include documentation files
.DBF	Database files
.NDX	Database index files
.XRF	Cross-reference files
.LOG	Log files
.LST	Task list files
.OBJ	Object files (BASIC, Control Block, or Configuration)
.POB	Drive parameter object files
.OS	System files
.EXE	” ”
.BAT	” ”
.INI	” ”
.ERM	” ”
.HLP	” ”
.IMA	” ”
.DAT	” ”

After you edit and save an existing file using the text editor or the PC editor, AutoMax renames the original file by adding the tilde character (~) at the beginning of the file extension and dropping the third letter in the file extension. This makes it possible to edit a file once without losing the original. For example, editing and then saving the task SPD.BLK would result in the edited version of the task being saved as SPD.BLK and SPD.~BL as the backup.

Note that if you choose a different text editor in AutoMax Setup (see section 5.5.2), the character used to designate backup files, if any, may be different than described above. If your text editor creates backup files by attaching a .BAK file extension, be sure that you give all task files unique names to avoid writing over an existing backup file. For example, if you create a BASIC task and an Include file with the same name (TASK1.BAS and TASK1.INC), and then edit TASK1.BAS, the backup file will be named TASK1.BAK. If you then edit the Include file, its backup would write over, and you would lose, the backup for the BASIC task.

The Executive software maintains a maximum of two versions of any task file. Backup copies of the database files are not maintained. If more than two versions of any file need to be maintained, the file should be copied under another name using the COPY command from the Windows File Manager.

4.21 Text Editor

The Norton Editor is part of the AutoMax Executive software, and is the default text editor used to create and edit Control Block and BASIC tasks, BASIC "include" files, and documentation files. The text editor is called up automatically whenever the task to be edited is a BASIC or Control Block task. See J-3618 for instructions on using the Norton Editor.

If you do not want to use the Norton Editor as your text editor, you can designate a different text editor. Note, however, that the Norton Editor is also used by the PC Editor (see section 4.22.5) to create remark sequences. Therefore, it must not be deleted from your hard disk even though you may be using another text editor. Use the following procedure to change text editors.

Step 1. Verify that the text editor is in your DOS PATH.

Step 2. If it is a DOS-based editor, you must also create a .PIF file for it. This is done with the Windows PIF Editor.

a. Access the Windows Program Manager.

b. Choose the Accessories Group.

c. Choose the PIF Editor.

d. Select Open from the File menu, and specify NE.COM.

e. In the Program Filename field, change NE.COM to the name of your text editor (as it appears in your DOS PATH).

f. In the Window Title field, enter the descriptive name of your text editor.

g. Save the file.

See the *Microsoft Windows User's Guide* for more information on the PIF Editor.

Step 3. Enter the name of your text editor during AutoMax Setup. See section 5.5.2 for the AutoMax Setup procedure.

4.22 PC Editor

The PC editor is used to create, edit, and view PC/Ladder Language tasks. The PC editor is automatically called up whenever you edit a PC task. Refer to J-3677 for more information on AutoMax Ladder Logic language.

When you are in the PC editor, the following keys have the special functions indicated:

Home	Displays the first sequence in the task
PgUp	Displays the previous sequence in the task
PgDn	Displays the next sequence in the task
End	Displays the last sequence in the task
Del	Deletes the sequence being displayed from the PC task. The next sequence in the program will be displayed unless the last sequence in the task was deleted, in which case the current last sequence will be displayed.
Ins	Inserts a sequence following the sequence being displayed. The new sequence number will be increased by 10 as long as there is no other sequence

number in that range. If there a sequence number in that range, the sequence number will be increased by one. If that is not possible, the system will display an error message.

The PC editor screen is a matrix of six rows by ten columns representing the space available for creating one sequence. The available sequence elements and file commands are displayed on the command line of the PC Editor screen. See figure 4.11 for a sample PC editor screen.

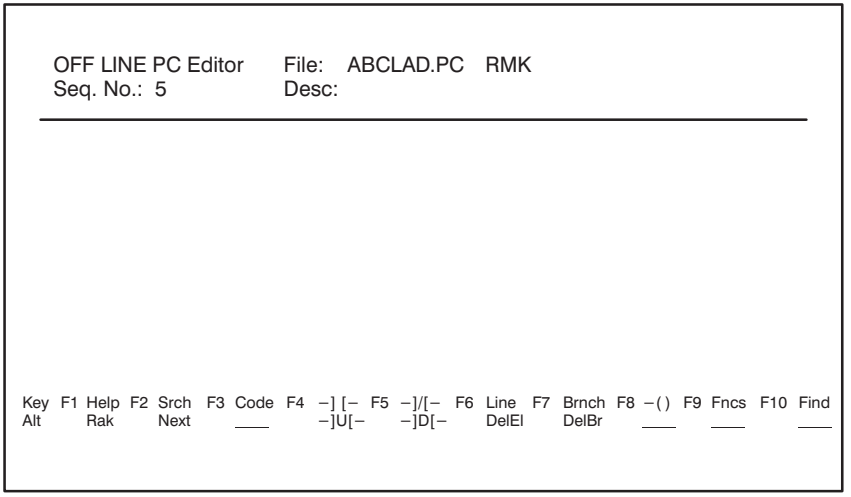


Figure 4.11 - PC Editor Screen

The PC editor functions and keystrokes required are described below. When two keys are shown separated with a slash, e.g., Alt/F2, they are to be typed simultaneously.

When you are creating a new Ladder Logic task, you will be prompted for the initial sequence number. After the first sequence, the editor will increment by 10 each time you press <PgDn>. When you edit an existing Ladder Logic task, pressing <PgDn> displays the next sequence in the task.

The PC editor opens a “journal” file of all keystrokes you entered for the current editing session. When you exit or quit the editor normally, the file is deleted. If, however, power to the personal computer is accidentally turned off, the journal file is saved and the user prompted the next time the PC editor is called up.

4.22.1 PC Editor Variable Names and Types

In AutoMax ladder logic, variable names are used in place of physical addresses. They are limited to 14 alphanumeric characters, and must begin with a letter. The editor automatically reserves two lines of seven characters each for each element. Typing the underscore “_” on the first line inserts an underscore and moves the cursor to the second line, allowing you to divide the variable name into recognizable segments.

Each variable name must be defined as one of two types: local to the task or common to all Processors in the rack. Local variables are those that do not represent a physical I/O point and are not referenced by any other application tasks running in the rack. Common variables are those that represent physical I/O points or are referenced by other tasks in the rack.

The PC editor will prompt you to enter "L" for local or "C" for common. If you enter "L", the variable will then be displayed in lower-case. If you enter "C", the variable will be displayed in uppercase. Inputs (contacts) are assumed to be common and displayed in uppercase by the editor unless you have already specified them as local earlier in the task. If you specify the variable as local later in the task, the variable will be displayed in lower-case when you edit or view the initial sequence containing the variable.

Each variable name can have a 40-character description which can be printed out along with the sequences. However, the PC editor only prompts you for a 40-character description for coils. Descriptions for other variables can be entered by using F3 A, described below. Only the coil description will be displayed on the screen when you edit a sequence. When you monitor PC tasks in the online mode (see 18.4), however, the screen will display the description for any variable you are monitoring.

The description is broken down into four groups of 10 characters. The " ^ " characters indicate where the line will be broken up on the printout of the task.

4.22.2 PC Editor File Commands

F3 is used to initiate a file command. F3 followed by one of the following keys executes the indicated command.

E - Exits the PC editor and saves updated file. If an existing file was being edited, the old file will be renamed "FILENAME.~PC".

Q - Quits the PC editor without saving. Any modifications to the task are lost.

R - Resequences a range of sequence numbers. This function allows you to renumber a portion or an entire task. You will be prompted for the first and last sequence number to be resequenced (inclusive), the new first sequence number, and the new increment.

M - Moves a sequence or range of sequences to another part of the task. You will be prompted for the first and last sequence numbers to move and the new first sequence number. If you are moving a set of sequences, they will be numbered using an increment of 10 after the first sequence. If an increment of 10 causes an overlap of sequence numbers, the editor will attempt to increment by 1. If overlap occurs, an error message is displayed.

S - Substitutes a new/existing variable name for an existing name. You will be prompted for a range of sequence numbers to which the substitution will apply. The substitution can be qualified to apply to certain element types only by entering one of the following options:

A for all contacts and coil
O for normally open contacts
C for normally closed contacts
U for upward transition contacts
D for downward transition contacts

P - Modifies the Preset Value for a Counter(s) or Timer(s). The unit is 1/10th of a second. Legal values range from 1 to 32767 for timers and +32767 to -32768 for counters. See 4.22.4 for more information on Preset Value.

A - Adds a description for a variable name. This function is used to enter a description for a variable that does not already have one. When you use this function, the editor automatically displays variables that do not have a description, one at a time, along with a field for entering the description. As soon as you enter the description and press <CR>, any other variable that does not have a description is displayed. Enter the description in response to screen prompts.

C - Changes the description for a variable name. After you enter the variable name whose description you want to change, the current description is shown in reverse video. Entering a <CR> retains the description. To change the description, use the arrow keys to position the cursor in the field and enter the new description. Enter a <CR> when finished.

If you enter a single wild card character (*) for the variable name, you can scroll through all variable names in the task, regardless of whether they have descriptions.

I - Allows you to enter and view task information, including task header (description), scan time, execution time, and program size. Execution time and program size are determined by the PC editor. You must enter the scan time (time within which the task must run) and the task header, if any. Scan time is measured in seconds. The actual time the task will run has a resolution of one tick, which defaults to 5.5 milliseconds.

V - Changes a variable from local (L) to common (C) or common to local. If the variable is currently a local, the field will show “C” as the default and vice versa. Use a <CR> to select the default or type over it.

D - Deletes a range of sequences. The function will prompt for the first and last sequence number to be deleted.

W - Allows wild card substitution for variable names. Only one wild card character (*) is permitted in both the old and new strings entered. The wild card character must be in the same position in both the old and new string.

N - Includes a range of sequences from another file. At the “filespec.” prompt, enter the name of the file, including drive and subdirectory specification if different from the current. “First Sequence No.” is the first sequence you want to include from the file. “Last Sequence No.” is the last sequence of the range you want to include. “New Sequence No.” is the sequence number you want to assign to the “First Sequence No.” in the current task. See the Move

function description (“M”) for errors that may also occur when you include sequences. Note that remark sequence text will not be moved to the current task.

T - Allows you to edit element descriptions in the current task using the Norton Editor. This function creates a temporary file with the extension .DTX constructed each time you invoke the function. Variable names in the task are shown on separate lines. Each variable is followed by a > character. Enter your element description after the >. Note that this function is similar to add (“A”) and change (“C”), but is more useful for making changes to more than one element description. Descriptions are limited to 40 characters.

4.22.3 PC Editor Input Operations

The following inputs are supported by the PC Editor.

F4 - ---] [--- Normally Open Contact

Places the contact at the current location of the cursor. Enter the name of the contact at the blinking cursor above the symbol.

Alt/F4 - ---]U[--- Upward Transition Contact

Places the contact at the current location of the cursor. Enter the name of the contact at the blinking cursor above the symbol.

F5 - ---]/[--- Normally Closed Contact

Places the contact at the current location of the cursor. Enter the name of the contact at the blinking cursor above the symbol.

Alt/F5 - ---]D[--- Downward Transition Contact

Places the contact at the current location of the cursor. Enter the name of the contact at the blinking cursor above the symbol.

F6 - -------- Horizontal Line

Places the line at the current location of the cursor, extending to the right one column in the ten-column matrix allowed for each element.

Alt/F6 - DelEl DeleteHorizontal Element

Deletes the element to the right of the cursor.

F7 - Br Vertical Branch

Places a branch at the current location of the cursor, extending down one row in the 6-row matrix allowed for each element.

Alt/F7 - DelBr Delete Vertical Branch.

Deletes the branch below the cursor.

4.22.4 PC Editor Output Operations

The AutoMax PC Editor supports the following output, or coil, operations. A variable name can only be used once as a coil in a ladder logic task. Enter “L” for local and “C” for common in response to the prompt for variable type.

F8 - -----( ) Coil

Enters a coil. The cursor must be in the top row of the matrix allowed for each ladder sequence. Enter the coil name above the symbol.

You will then be prompted for a coil description which begins at the blinking cursor. The description field will show the last coil description entered as the default.

F9 - Other Operations

Five other output operations are described below. They are entered using the F9 key followed by the letter indicated. To enter one of these outputs, the cursor must be in the top row of the matrix allowed for a ladder sequence.

1. T - On-Delay Timer Coil

The On-Delay Timer provides a logically true output whenever the input is true and the preset time interval has elapsed.

Enter the coil name and define it as either local or common. Then enter the coil description. You will then be prompted for the “PN”, “PV”, and “CN”.

The PN, or preset name, is the name of the variable that specifies the delay between the input turning on and the output turning on. You must define the PN as either local or common.

Next, enter the PV, or preset value. The preset value unit is 1/10th of a second. For example, a PV of 10 corresponds to a 1.0 second delay. Legal values range from 1 to 32767.

The CN, or current value name, is the name of the variable that specifes the current value of the timer. Current names must be unique. After you enter the CN, define it as either local or common.

The CV, or current value, is not specified at this time. It will be displayed only when you are online and monitoring the task.

2. O - Off-Delay Timer Coil

The Off-Delay timer provides a logically false output whenever the input is false and the preset time interval has elapsed.

Enter the off-delay timer coil in the same manner as the on-delay timer coil described in number 1 above. With the Off-Delay Timer Coil, the preset value (PV) defines how long the output stays on after the input turns off.

3. C - Counter Coil

The Counter is capable of counting over the range of +/-32767.

Enter the coil name and define it as either local or common. Then enter the coil description. You will then be prompted for the “PN”, “PV”, and “CN”.

The preset name (PN) is the name of the variable that selects the value at which the output will turn on. Define the PN as either local or common.

Next, enter the preset value (PV). Legal values range from -32767 to +32767.

The current value name (CN) is the name of the variable that holds the current value of the counter. After entering the CN, define it as either local or common.

The current value (CV) is not specified at this time. It is displayed only when you are online and monitoring the task.

4. S - Shift Register Coil

The Shift Register has a variable length of up to 16 bits and shifts to the right.

Enter the shift register coil name and define it as either local or common. Then enter the coil description. You will then be prompted for “LEN” and “CN”.

“LEN” specifies the number of bits in the shift register. Legal values range from 1 to 16.

The current value name (CN) is the name of the variable that holds the current value of the shift register. After you enter the CN, define it as either local or common.

The current value (CV) is not specified at this time. It is displayed only when you are online and monitoring the task.

5. E - Set Event Coil

The set event coil synchronizes the operation of other tasks with the present task by setting the event name true. Enter the coil name and define it as either local or common. Next, enter the coil description.

Enter the event name at the “EN” prompt. The event will be set true when the rising edge of the input is detected.

4.22.5 PC Editor REMARK Sequence

F9 followed by R is used to reserve up to 16 80-character long lines of remark text as a “sequence”. You need to enter only the remark keyname at the desired sequence. Remark keynames can be up to 14 characters long and must begin with a letter. You may use letters, numbers, or the underscore character (_) within the keyname. All remark keynames in the application task, preceded by a “!” (added by the PC editor), are written to a separate file with the same name as the task, but with the extension .REM.

Method 1

There are two methods of entering remark text. The first method is the default mode (remarks text enabled, RMK displayed in the top line). When you enter a remark keyname, the Norton editor will be invoked. You can then enter up to 16 80-character lines immediately underneath the keyname. When you exit the Norton editor, the keyname and text are added to the .REM file.

Method 2

The second method of entering remark text is to disable the remark text input/display by toggling ALT/F1. In this mode, the PC editor will accept only a keyname for a remark sequence.Because the remark must be entered through a text editor, you must exit the PC editor using F3 E for exit in order to enter the text for the remark. You are not, however, required to exit the PC editor after entering a remark keyname. You can enter the text for all remark sequences at one time when you are finished editing the task.

Whenever you exit the PC editor and there are remark keynames in the task, AutoMax calls up the Norton editor to allow you to enter or edit the text for each remark in the task. The keynames are stored in the .REM file in the order they are found in the application task, not in the order they are entered. Enter the remark text on the line following the remark keyname.

Note that remark keynames may not be modified. They may only be deleted and then a new keyname and text entered.

4.22.6 Searching for a Sequence or Coil Variable Name

F10 from the PC editor command line is used to find a sequence number or coil variable name. To find a variable name or sequence number, enter the name of the coil or the sequence number at the prompt and the task will be searched for the sequence or coil name. If the sequence is not found, the PC editor will initiate an insert operation.

4.22.7 Searching for a Variable Name

F2 from the PC editor command line initiates a search for a variable name that does not represent a coil. You can qualify the search by entering the following options:

A	All contacts and coil
O	Normally open contacts
C	Normally closed contacts
U	Upward transition contacts
D	Downward transition contacts

The task will be scanned from the beginning for a match. If a match is found, the search will stop and the sequence containing the match will be displayed.

Alt/F2 is used to search for the next match. A search must have been initiated with the F2 function key in order to use Alt/F2. The search will continue from the sequence where it had stopped and go on to the next match. This function may be re-executed as many times as necessary.

4.22.8 Inserting a New First Sequence

F10 from the PC editor command line is used to insert a new first sequence. At the prompt, enter a sequence number smaller than that of the current first sequence. Then enter your sequence. This is the only method of inserting a sequence before the current first sequence in the task.

5.0 USING THE SYSTEM CONFIGURATOR

The System Configurator is used to create, edit, import, and delete systems, sections, and racks. The System Configurator is the first screen displayed after the AutoMax Programming Executive identification screen. Three columns entitled Systems, Sections, and Racks are shown on the screen.

These columns list in alphabetical order the systems in the current library, sections in the selected system, and racks in the selected section.

The current library is displayed on the top line of the screen. A library is the highest level of organization in the AutoMax Programming Executive. Normally, the first system in the list will be highlighted as the default selection, along with the first section and rack in the two other columns. The sections displayed are sections within the selected system. Likewise, the racks displayed are racks within the selected section. Figure 5.1 illustrates the System Configurator menu structure.

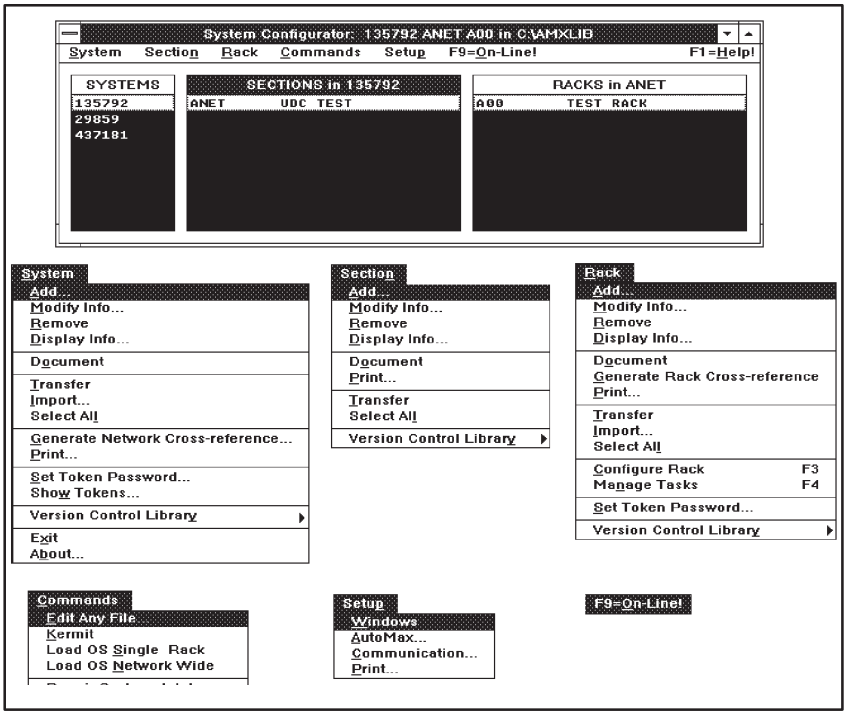


Figure 5.1 - System Configurator Menus

5.1 System Menu

The System menu includes all of the commands that operate on systems. A system is the (upper level) designation for an entire application, e.g., a plating line system. You can add new systems or import existing ASD Version 8.x systems. You can modify, edit, transfer, and remove existing systems, as well as print the system files. The sections that follow describe the commands in the order in which they appear on the System menu. For each of the commands, required data fields are denoted by (R) and optional data fields are denoted as (O). See Appendix M for information on copying systems.

5.1.1 Adding a System

You can add a new system by naming the system and then adding sections and racks. Use the following procedure to add a new system. See figure 5.2 for an illustration of the Add System dialog box.

Step 1. Select Add from the System menu to display the Add System dialog box.

Step 2. Enter the following information:

System name - The application name; an alphanumeric string of up to 8 characters (no spaces, hyphens, or underscores). The system name must be unique within the current library. (R)

Customer - The end user of the system; an alphanumeric string of up to 40 characters. (O)

Engineer - The engineer in charge of the system; an alphanumeric string of up to 31 characters.(O).

Description - A description of the system; an alphanumeric string of up to 40 characters. (O)

Step 3. Select OK to add the system to the database or Cancel to return to the System Configurator without adding the system. When a system is added, it will be inserted into the list of systems in alphabetical order and become the selected system.

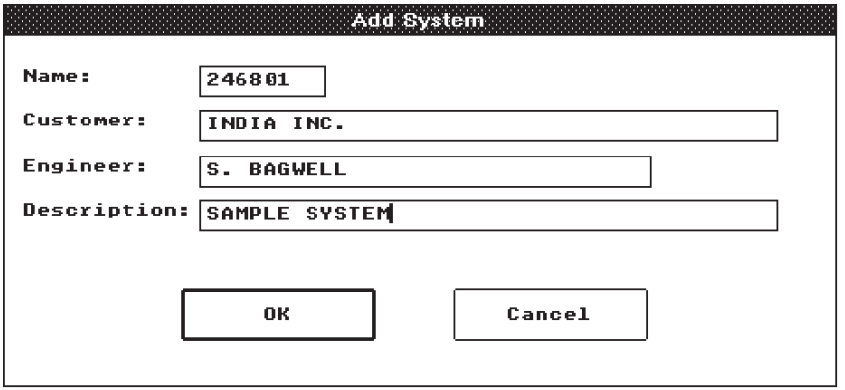


Figure 5.2 - Add System

5.1.2 Modifying System Information

Select Modify Info to display the Modify System dialog box for the selected system. You can make changes to information displayed in the Customer, Engineer, or Description fields. The Name field cannot be changed. If there is an error in the name of a system, you must remove the system, then re-enter it using Add. Select OK to save the changes to the database or Cancel to return to the System Configurator without modifying the system information.

5.1.3 Removing a System

Select Remove to delete one or more selected systems, including all the associated sections, racks, and tasks. A confirmation box will appear on the screen.

Note carefully that the selected system and all its sections, racks, modules, tasks, and variables will be deleted if you confirm the remove command with "OK".

5.1.4 Displaying System Information

Select Display Info to display the Display System dialog box for the selected system, which will list the name, customer, engineer, and description fields for the system. You cannot make changes to any of the information displayed. Select OK to return to the main window.

5.1.5 Editing the Documentation File for a System

Select Document to access the text editor, which you can use to create or edit the documentation file associated with the selected system. The file will be named automatically (the same name as the system, with an .SDC extension) and will be stored in the system subdirectory under the library directory. See section 4.21 for instructions on using the text editor.

5.1.6 Printing System Files

The Print System command enables you to print out files relevant to one or more systems. After selecting one or more systems from the System list, select Print from the System menu. A dialog box with the various print options will be displayed (see the figure below). Note that “cross reference” is abbreviated “xrf” in some dialog boxes.

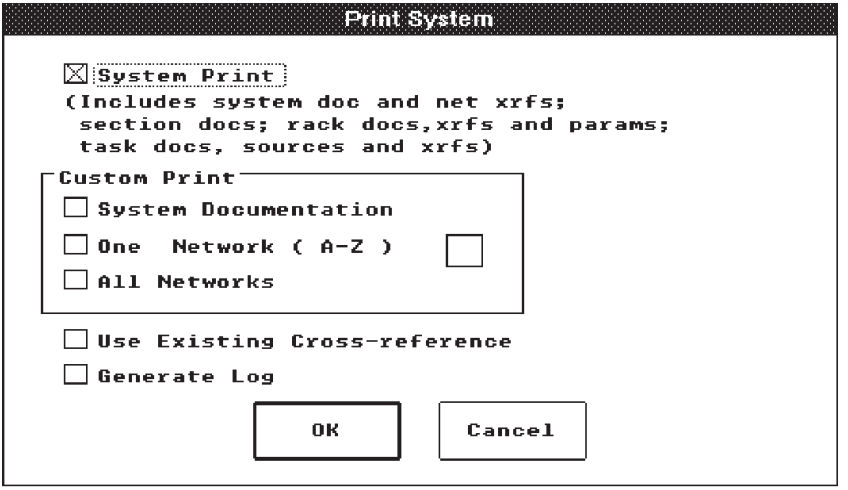


Figure 5.3 - System Print Dialog Box

There are several print options at the system level. Options are enabled (or disabled) depending on which check boxes are selected (checked). The dialog box will initially display System Print as selected (checked). The Use Existing Cross Reference Files and Log Errors options will be enabled, but not selected. The other options will be disabled.

Selecting System Print will print out the following for each of the selected systems:

- System title page (system name, description, engineer, customer, and a list of the sections in the system.
- System documentation file
- Network cross reference file
  - Section title page (section name, description, engineer, and a list of racks in the section).
  - Section documentation file
    - Rack title page (rack name, description, engineer, location, and a list of the tasks in the rack).
    - Rack documentation file
    - Rack cross reference file

- Rack parameter file (only for racks containing UDC modules)
    - Task title page (task name, description, engineer, slot, priority, critical, reconstructible, utility).
    - Task documentation file
    - Task source code
    - Task cross reference file

If you de-select (uncheck) System Print, the System Documentation and Network Cross Reference options (listed in the Custom Print box) will be enabled. Select **System Documentation** to print out the system documentation file.

The Network Cross Reference options are mutually exclusive. If you choose **One Network**, you must also enter the network letter. If multiple systems have been selected, the One Network option will be disabled. If you choose **All Networks**, the network cross reference will be printed for all of the networks in the system.

Select **Use Existing Cross Reference Files** if you know that the cross reference files are current. Selecting this option will save time since new files will not be generated before printing. If cross reference files do not already exist, they will be generated and printed. Note that cross reference files generated by the print routine will be deleted after printing.

Select **Generate Log** to create a log file that will list any errors that may occur during the print job. Each system will have a separate log file. The log file will exist under the system directory and will be named _PRINT.LOG. If there are no errors, the log file will simply list the files that were printed.

5.1.7 Transferring a System

The Transfer command can be used at the system, section, rack, or task level. The description that follows applies to Transfer at all levels.

The Transfer command is used to transfer copies of all or part of an AutoMax system from one computer to another, e.g., from the development site where the system is configured and application tasks are written to the actual installation site via floppy disks or removable hard disks.

When using the Transfer command to provide the required files for a site which is using the Run-Time Executive software, the supplier should select the “Object Files” option described in the procedure below. This will provide the target site with all the necessary files (configuration, task, and drive parameter files) as well as the database files. Note that the database files are not required when using the Run-Time Programming Executive software, and can be deleted from the diskettes that will be provided to the site.

What is Transferred?

The information that the Executive software keeps for an entity (system, section, rack, or task) is considered to be part of that entity when transferring. This means that when a rack, for example, is transferred, its description, location, and engineer may be changed at the destination, and this information will be brought back when the rack is transferred back to the site manager.

Transfer Options

Three different options are available to select the files that will be transferred in or out of the library. The “Task object files” and “Cross reference files” options can be selected when you transfer files, but this is not necessary because they can be re-generated from the task files. The “System wide information only” option transfers all files that define the system. It should not be used when transferring an entire System because all of the files in the system will typically be required at the eventual “destination” of the system. Task Transfer (from the Task Manager) can be used whenever a copy of one or more tasks (but not the whole rack) is desired.

Using Transfer for Backup

For medium and large systems, one personal computer at the installation site is typically used to store the entire system. The operator in charge of this personal computer will be referred to as the “site manager.” The site manager maintains an up-to-date copy of the entire system, and performs such functions as daily backup to a permanent storage medium for safe-keeping in the event of a hard disk failure. Backup also serves as a means to undo changes that do not work as intended. The Transfer command can be used to make these backup copies if a commercial backup package is not available.

A Typical Scenario

The software for individual racks is typically transferred to a portable personal computer which will be taken out to the rack to load, run, and de-bug the application tasks.

To transfer a rack or group of racks to the portable personal computer, the site manager would use the Transfer command on the Rack or Section menus, depending on how many racks are needed. Regardless of how many racks the portable contains, it will require a copy of the system-wide information. Without this information, AutoMax Executive software running on the portable would not be able to access the racks. To obtain a copy of the system-wide information, the site manager would need to run Transfer from the System menu, and select the “System-wide information only” option. The site manager may wish to consider making sets of the system-wide information floppies available whenever this information changes.

Once the racks and system-wide information are on floppies, the site manager will need to transfer them to the portable personal computer. To do this, the site manager should run the Executive on the portable and select Transfer from the System menu. Using the “System-wide information only” option, he should transfer the system information from the floppies to the hard disk. Once the system information is there, he can select the system, section, and rack(s) desired, and use Transfer from either the Section or Rack menus to transfer the rack files from the floppies.

To transfer the racks back to the site manager for backup, printing, network updates, etc., an operator should Transfer from the Section or Racks menu to transfer the files, first to floppies, and then to the site manager. The system-wide information should never be transferred from the portable personal computer to the site manager.

Some Precautions

When racks and copies of the system-wide information are distributed to various personal computers, all users who have copies of the same files must be aware that only one of the copies can be considered to be read/write. All other copies are to be considered read-only. Since the system-wide information is used by all racks, it should only be changed when the entire system is present on the hard disk. For this reason, the copy of the system-wide information at the site manager should be considered the only changeable copy. This means that no changes to the system structure, such as adding or removing sections and racks, should be made on the portable personal computers. No changes to network-wide variables should be made on the portable personal computers because these are also part of the system-wide information. Any network changes that are required for de-bugging on the portable personal computers should be done with unlinked local variables. These changes must later be made network-wide when the rack is transferred back to the site manager's PC. When a rack is transferred from the site manager to a portable personal computer for debugging, the copy at the site manager must be considered read-only.

Step-by-Step Procedure for System Transfer

Use the following procedure to transfer systems. If you are transferring to diskettes, you will need at least one diskette for each rack in the system. You will be prompted for additional diskettes, if needed. Diskettes can be formatted as needed or, to save time, you can use blank pre-formatted diskettes. Figure 5.4 illustrates the System Transfer dialog box.

Step 1. Select the system(s) to be transferred from the systems list.

Step 2. Select Transfer from the System menu. The Transfer dialog box will be displayed with the current AutoMax Executive drive and library (as designated in AutoMax Setup) listed in a box on the top left.

Step 3. Enter information for the following fields. Some fields may contain default entries. If you do not change these fields, the default will be used.

Direction arrows - Transfer the system **To** (left arrow) or **From** (right arrow) the current AutoMax Executive V3.x library. Select the appropriate option button.

Drive - Enter the source drive or the destination drive. If A or B is entered, AutoMax Executive V3.x assumes the drive is a floppy drive; otherwise, AutoMax assumes a hard drive.(R)

Library - Enter the library directory to copy the system(s) from or to. (R)

Format Floppies - Select if you want to format destination diskettes.

High Capacity - Select if you want to format and/or write to high density diskettes. The appropriate diskette capacity will be displayed.

System(s) - The AutoMax Executive will list the system(s) selected when the Transfer command was called. If you are transferring out of the current library, make certain that this field lists all of the systems you have selected. If you

are transferring systems in, enter the system names separated by commas or spaces. (R)

System-wide information only - If this box is selected, only the system, section, rack, and network databases, and the system documentation file will be transferred.

Step 4. Select which of the following groups of files you want to transfer with the system. These files are not necessary since they can be regenerated from the source files.

Object files - the compiled application tasks, parameter object files, and rack configuration object file(s).

Cross-reference files

Transfer Tokens (See Appendix N for more information.)

Step 5. Select Help for information about the transfer procedure, Cancel to return to the System Configurator without transferring a system, or OK to begin the system transfer.

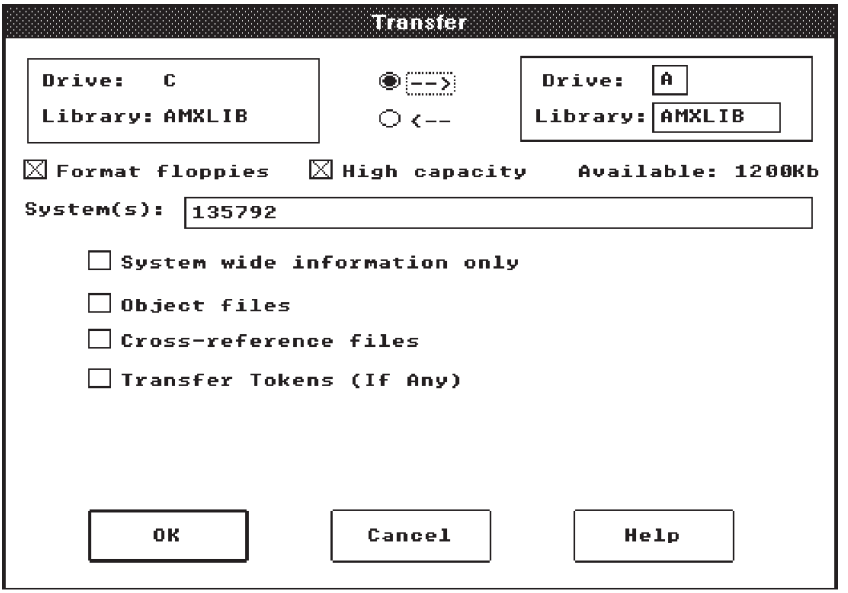


Figure 5.4 - Transfer System

5.1.8 Importing a System from ASD

You can import a system created with the AutoMax Software Designer (ASD) Version 8.x (M/N 57C310-57C313) into the AutoMax Executive software. The system must not already exist in the AutoMax Executive software. This procedure does not delete, or alter in any way, the original ASD system.

The Import procedure will add the system, sections, and racks using information stored in the ASD file. One section is created for each network, and all of the racks named with that network's letter are stored in that section. If a rack is connected to more than one network, the "primary" network is used. The rack's name in ASD (for

example, A00) indicates which network is primary (in this example, A). The description, engineer, and customer data for the system and each rack are transferred from the ASD file. Import adds racks using the same process as for importing an AutoMax Version 2 rack.

The Import procedure will add as many modules to the rack as possible based on the definition type found for a given slot (see section 5.1.8.1 for information regarding module types). Import will add the variables and their descriptions defined in the configuration task. The network-wide variable names will be added using the .NET files, if any. .NET files define network addresses for GBLDEF variables in configuration tasks.

The tasks (name, type, priority, critical flag, and Processor slot) defined in the configuration task will also be added. Task source files will be copied into the AutoMax Executive system. Note that tasks must successfully compile in ASD in order to import properly. Since BASIC include files (.INC) are treated as tasks in the AutoMax Executive software, a task is added for each include file found. Import will also copy and rename any system, rack, or task documentation files from the ASD system into the AutoMax Executive system. The CPU documentation files in the ASD system will be copied into AutoMax Executive system, but they can be accessed by the AutoMax Executive software only by using Edit File from the Commands menu. You may want to copy pertinent information into the documentation files for the appropriate racks.

5.1.8.1 Module Types Recognized by the Import Procedure

Import will add as many modules as possible from the variable definition statements in the configuration task. The following list shows the modules added for the type of definition found.

Definition Found	Type of Module Added	
NETDEF or GBLDEF	57C404	Network Communications
MODDEF	57C414	Modbus interface
RIODEF	57C416	Remote I/O Communications (in the master rack)
RNETDEF	57C417	AutoMate interface
ABDEF	57C418	AB interface
TASK	57C430	AutoMax Processor
IODEF		Generic module (unless one of the other definitions is found for the slot)

5.1.8.2 Generic Modules

A “generic” module, identified by the prefix “GEN”, is added to the appropriate slot in the local rack when IODEF statements are the only statements found for the slot in the configuration task. The generic module is used because the IODEF statement does not provide enough information to determine the specific type of module used. Note also that for a remote I/O network, no hardware is added to the rack database for the same reason.

5.1.8.3 Accessibility of Variables Added by Import

Although the Remote I/O hardware which was configured in the configuration task does not get added to the rack, all variables defined for it are added to the rack's database. In the local (master) rack, the variables defined for the corresponding slot will be mapped to a generic module. Remote I/O variables are not accessible to the user until the correct module or rail is actually added to the Remote rack or Head through the Rack Configurator.

Variables that exist in the rack database, but are not attached to any module at all, are called floating variables. These variables must be either deleted from the databases or the correct modules added to account for them and to properly document your application. Note carefully that when you configure the remote I/O network hardware in the Rack Configurator after the import procedure, you must be careful to add the correct hardware (proper size rack or Remote Head) the first time. This is important because if you make a mistake and add the wrong hardware, the only method of modifying the configuration is to remove the rack or Head and add the correct rack or Head. However, removing the rack or Head will also delete all of the floating variables that are mapped to the remote I/O system. When you configure the remote I/O system, you will be prompted that variables have already been configured for the hardware.

If floating variables exist in a remote I/O network when the configuration object file is generated in the Task Manager, they will be included in the configuration object code even though they are not accessible in the Variable Configurator. Floating variables in the master rack are flagged as errors. You can compile tasks when there are floating variables in a remote I/O network. You cannot, however, compile tasks when there are floating variables in a master rack.

Local network variables will be linked to network-wide names whenever possible. If there is a network-wide name for the network point used by a variable, the network-wide variable is linked to that variable. Any network or remote I/O registers/ points configured as both single and double integers will not import correctly. The procedure will result in prompts that indicate "duplicate Multibus addresses" and will require you to delete the offending variables.

5.1.8.4 Register Assignments on Modules Added by Import

Registers defined for some modules in AutoMax V2 cannot be configured in AutoMax Executive V3.x and will cause an error message to be displayed (invalid/incompatible data type) when you try to configure the module if you import the system or rack without first checking your AutoMax V2 configuration. Refer to Appendix K for examples.

Also, you can not define single and double integers in the same addresses (except for generic cards in both local and remote racks, where you have both an integer view and a double integer view). In version 2.0, defining an address as both integer and double integer was sometimes done in BLOCK_MOVE (BASIC) commands. When you try to configure a module (created through the Import procedure) that has integers and double integers defined in the same register address, a warning will appear in a dialog box that tells you that duplicate addresses were found. Double integer variables will be loaded into the database for the module first, so the duplicate address will always refer to the single integer variable. The dialog box will display the name of the integer variable and the register number

and prompt you to delete it. A dialog box will be displayed for each duplicate variable.

To fix this error, first write down the variable name and register. Then, click on OK to delete the variable. If you do not delete the offending variables, you will not be able to configure the module. When you have deleted all the offending variables and noted all the variable names and locations, configure the module as desired. You may have to edit your application tasks to reflect the new configuration.

5.1.8.5 Source Drive and Subdirectory for Import

It is important that the files to be imported be stored in a specific manner as follows. The only configuration task in the subdirectory should be the configuration task for the rack you want to import. The configuration task does not need to be compiled because the import procedure will compile it. If there is more than one configuration task in the subdirectory, the import procedure will use the first one found, which may not necessarily be the configuration you want to import.

The Import procedure will copy in all the application tasks (.BAS, .BLK, .PC. and .INC) found in the subdirectory. However, only those tasks with corresponding TASK statements in the configuration task will be added to the tasks database. If no application task exists for a TASK statement, you will be prompted to delete the task when you enter the Task Manager. Utility tasks (those without TASK statements) can subsequently be added in the Task Manager, which can use the imported source file of the same name.

5.1.8.6 Procedure for Importing an ASD System

Use the following procedure to import a system (see the AutoMax Software Designer instruction manual (J-3615) and the Kermit Reference manual (J-3616) for steps 1 through 3). Figure 5.5 illustrates the Import ASD System dialog box.

Step 1. Using ASD, use the Verify command to ensure the ASD files are correct.

Step 2. Using ASD, download the ASD V8 system from the VAX/VMS to the personal computer.

Step 3. Using the Kermit communications software, download the ASD file.

a. Set the working directory on the personal computer to <d>:\<system>.

b. Use the Kermit SERVER command to put the VAX into server mode.

c. Use the GET command as follows: GET ASD_LIB:[system_name]system_name.ASD

Step 4. Select Import from the AutoMax System menu.

Step 5. Enter the following information:

Source drive - Enter the letter of the drive into which you downloaded the ASD system to be imported into the AutoMax Executive software.

Source system name - Enter the name of the ASD system to be imported.

Step 6. Select OK to import the system or Cancel to return to the System Configurator without importing the system. If a system is imported, it will be inserted into the list of systems in alphabetical order and become the selected system.

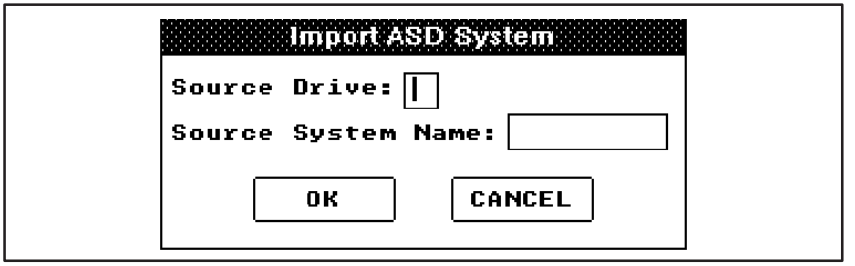


Figure 5.5 - Import ASD System

5.1.9 Selecting All Systems

The Select All command selects all the systems listed. Select All can be used when you want to transfer, print documentation files for all systems, or remove all of the systems. To select one system after selecting all, simply click on the desired system.

5.1.10 Generating Network Cross Reference Files

The network cross reference lists all the network variables used in the system organized by network/drop/register/bit. It also provides a list of local variables not linked to network-wide variables, network registers which are read but not written to, and network registers that are written to but not read. You can generate a network cross reference for one network in a system, all the networks within a single system, or for all networks in multiple systems. The fields described below are listed across the top of the network cross reference printout. See figure 5.6 for an illustration of the network cross reference.

Network Cross Reference SYSTEM: NETXREF NETWORK: A PAGE: A1

H	D	R	B	Network-wide Name	Written as	in rack	Read as	in rack	Description
A	01	00		G_A010%					Drop Area 1 Register 0
A	01	01		G_A011%	L_A011%	A01	L_INT1% L_A011%	A02 A00	Drop Area 1 Register 1
A	01	02			L_A012%	A01	L_A012% L_INT2%	A00 A02	Drop Area 1 Register 1

Network Cross Reference SYSTEM: NETXREF NETWORK: A Local Var. not linked to Network-wide var. PAGE: A2

H	D	R	B	Network-wide Name	Written as	in rack	Read as	in rack	Description
A	01	01		G_A011%			L_INT1%	A02	Drop Area 1 Register 1

Figure 5.6 - Network Cross Reference

N - the alphabetic character that designates the network

D - the drop number

R - the register to which the variable is assigned

B - the bit number, if any, to which the variable is assigned

Network-wide Name - the network-wide name of the variable

Written as - the name given to the variable by the rack(s) that write to it

in rack - the name of the rack that writes to the variable

Read as - the name of the variable in the racks that read it

in rack - the name of the rack that reads the variable

Description - the variable description, if given anywhere

Select one or more systems from the System list. Then select Generate Network Cross Reference from the System menu. The Generate Network Cross Reference dialog box will be displayed.

Select **One Network [A-Z]** to generate a cross reference file for a single network. Enter the letter that identifies the network. This option will be disabled if multiple systems were selected.

Select **All Networks** to generate a network cross reference file for all the networks in the selected system(s).

5.1.11 Using System Tokens and Token Passwords

If you are using tokens to protect your database files, the commands "Set Token Password" and "Show Tokens" will appear on the System menu. Refer to Appendix N for descriptions of how to use these commands.

5.1.12 Using Version Control Library

If your AutoMax system is using the Version Control Library facility, the command "Version Control Library" will appear on the System menu. Refer to Appendix P for descriptions of how to use the VCL commands.

5.1.13 Exiting AutoMax Executive V3.x

Selecting Exit from the System menu will exit the System Configurator. You will exit to either the Windows Program Manager or to the DOS prompt (as designated in AutoMax Setup).

5.1.14 About

Select About to display version and copyright information about the AutoMax Executive software and the DPS drive software, if installed.

5.2 Section Menu

The Section menu is used to add, modify, edit, print documentation files for the section, transfer, and remove sections in a system. Recall that a section is a collection of racks within a system. Sections are used to organize the system according to a criterion such as function or location. You must create at least one section for every system. The sections that follow describe the commands in the order in which

the commands appear on the Section menu. For each of the procedures, required data fields are denoted by (R) and optional data fields are denoted as (O).

5.2.1 Adding a Section

Use the following procedure to add a section. Figure 5.7 illustrates the Add Section dialog box.

Step 1. Select Add to display the Add Section dialog box.

Step 2. Enter the following information:

Section Name - A descriptive name (unique within the system) for a group of racks; an alphanumeric string of up to 8 characters (no spaces, hyphens, or underscores). (R)

Engineer - Enter the name of the engineer in charge of the section. (O)

Description - A description of the section; an alphanumeric string of up to 40 characters. (O)

Step 3. Select OK to add the section to the database or Cancel to return to the System Configurator without adding the section. When the section is added, it will be inserted into the list of sections in alphabetical order and becomes the selected section.

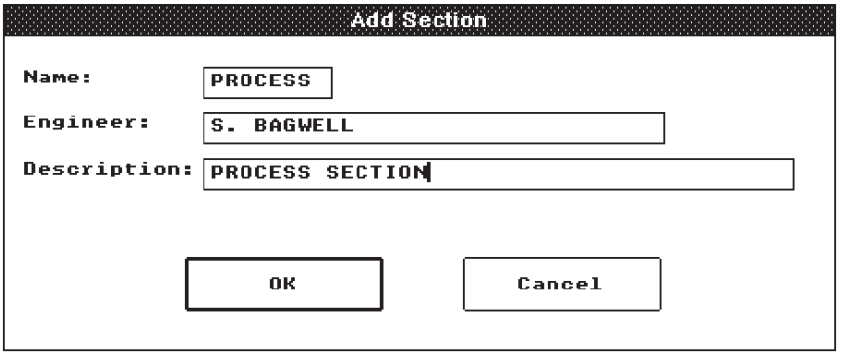


Figure 5.7 - Add Section

5.2.2 Modifying Section Information

Select Modify Info to display the Modify Section dialog box. You can make changes to any information except the section name. If there is an error in the name of a section, you must remove the section, then re-enter it using Add. Select OK to save the changes to the database or Cancel to return to the System Configurator without modifying the section information.

5.2.3 Removing a Section

Select Remove to delete one or more selected sections, including all the associated racks and tasks. A confirmation box will appear on the screen. Note carefully that the selected section(s) and all associated racks, modules, tasks, and variables will be deleted from the database if you confirm the remove command with “OK”.

5.2.4 Displaying Section Information

Select Display Info to display the Display Section dialog box, which lists the section name, engineer, and description. You cannot make changes to any of the information displayed. Select OK to return to the System Configurator.

5.2.5 Editing the Documentation File for a Section

Select Document to access the text editor, which you can use to create or edit the documentation file associated with the selected section. The file will be named automatically (the section name with a .CDC extension) and will be stored in the system subdirectory under the library directory. See section 4.21 for instructions on using the text editor.

5.2.6 Printing Section Files

The Print Section command enables you to print out files relevant to one or more sections of a system. After selecting one or more sections from the Section list, select Print from the Section menu. A dialog box with the various print options will be displayed (see the figure below).

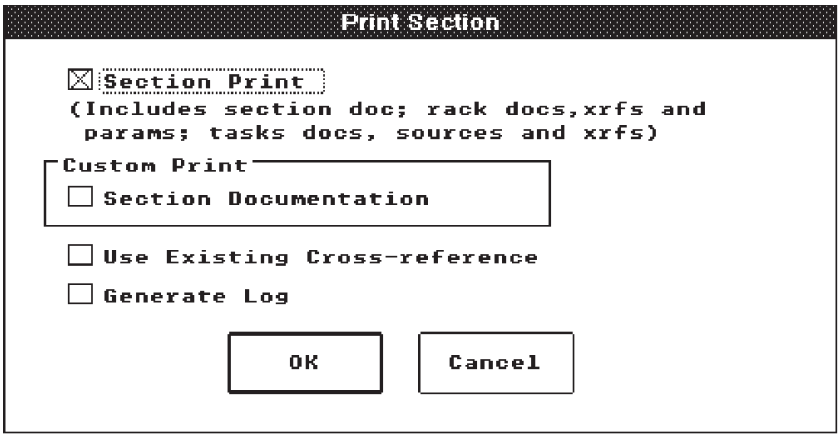


Figure 5.8 - Print Section Dialog Box

Selecting Section Print will print out the following for each of the selected sections:

- Section title page (section name, description, engineer, and a list of racks in the section).

- Section documentation file
  - Rack title page (rack name, description, engineer, location, and a list of the tasks in the rack).
  - Rack documentation file
  - Rack cross reference file
  - Rack parameter file (only for racks containing UDC modules)
    - Task title page (task name, description, engineer, slot, priority, critical, reconstructible, utility).
    - Task documentation file
    - Task source code
    - Task cross reference file

If you de-select (uncheck) Section Print, the Section Documentation option will be enabled. Select Section Documentation to print out the section documentation file.

Select **Use Existing Cross Reference Files** if you know that the cross reference files are current. Selecting this option will save time since new files will not be generated before printing. If cross reference files do not already exist, they will be generated and printed. Note that cross reference files generated by the print routine will be deleted after printing.

Select **Generate Log** to create a log file that will list any errors that may occur during the print job. Each section will have a separate log file. The log file will exist under the system directory and will be named _PRINT.LOG. If there are no errors, the log file will simply list the files that were printed.

5.2.7 Transferring a Section

You can copy all the files for one or more selected AutoMax Executive sections to diskettes (or another hard drive, or another library on the same drive), or copy all the files for an AutoMax Executive section from diskettes (or a hard drive) into the current AutoMax library (see section 5.1.7 for additional information). If you are transferring to diskettes, you will need at least one diskette for each rack in the section. You will be prompted for additional diskettes, if needed. Diskettes can be formatted as needed or, to save time, you can use blank pre-formatted diskettes. Use the following procedure to transfer sections. Figure 5.9 illustrates the Transfer Section dialog box.

Step 1. Select the section(s) to be transferred from the Section list.

Step 2. Select Transfer from the Section menu. The Transfer dialog box will be displayed with the current AutoMax Executive drive and library (as designated in AutoMax Setup) listed in a box on the top left.

Step 3. Enter information for the following fields. Some fields may contain default entries. If you do not change these fields, the default will be used.

Direction arrows - Transfer the section To (left arrow) or From (right arrow) the current AutoMax Executive library. Select the appropriate option button.

Drive - Enter the source drive or the destination drive. If A or B is entered, the AutoMax Executive software assumes the drive is a floppy drive; otherwise, AutoMax assumes a hard drive.

Library - Enter the library directory to copy the section(s) from or to.

Format Floppies - Select if you want to format destination diskettes.

High Capacity - Select if you want to format and/or write to high density diskettes. The appropriate diskette capacity will be displayed.

Section(s) - The Executive software will list the sections(s) selected when the Transfer command was called. If you are transferring out of the current library and system, make certain that this field lists all of the sections you have selected. If you are transferring sections in, enter the section names separated by commas or spaces.

Step 4. Select which of the following groups of files you want to transfer with each section:

Object files - compiled application tasks, parameter object files, and rack configuration object files.

Cross-reference files

Transfer Tokens (See Appendix N for more information.)

Step 5. Select Help for information relative to the transfer procedure, Cancel to return to the System Configurator without transferring a section, or OK to begin the section transfer.

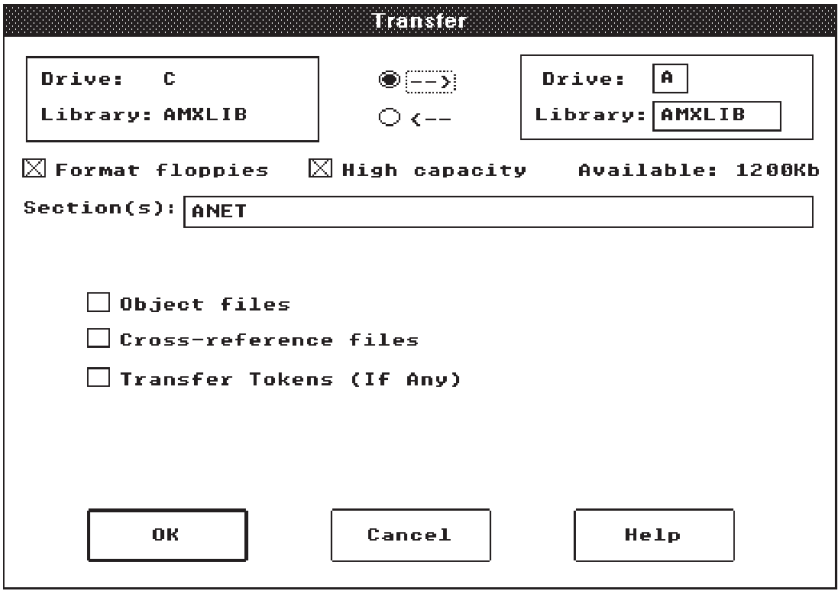


Figure 5.9 - Transfer Section

5.2.8 Selecting All Sections

The Select All command selects all the sections listed for the selected system. Select All can be used when you want to transfer, print documentation files, or remove all of the sections for a selected system. To select one section after selecting all, simply click on the selected section.

5.2.9 Using Version Control Library

If your AutoMax system is using the Version Control Library facility, the command “Version Control Library” will appear on the Section menu. Refer to Appendix P for how to use the VCL commands.

5.3 Rack Menu

The Rack menu is used to add, modify, edit, print the files for the rack, transfer, and remove racks in a section. Each rack corresponds directly to an AutoMax rack which is installed for the application.

The Rack menu provides access to the Rack Configurator, from which you can access the Variable Configurator. The Rack menu also provides access to Task Manager, from which you create/edit application tasks for the rack. The Rack Configurator, Variable Configurator, and Task Manager are described in sections 6, 7 and 8.

The sections that follow describe the commands in the Rack menu in the order that the commands appear on the menus. For each of the commands, required data fields are denoted by (R) and optional data fields are denoted as (O). See Appendix M for information about copying racks.

5.3.1 Adding a Rack

Use the following procedure to add a rack. Figure 5.10 illustrates the Add Rack dialog box.

Step 1. Select Add to display the Add Rack dialog box.

Step 2. Enter the following information. Some fields may contain default entries. If you do not change these fields, the default will be used.

Rack name - An alphanumeric string (unique within the system) of up to 8 characters (no spaces, hyphens, or underscores). (R)

10-slot or 16-slot rack - Select the appropriate type of rack.

Location - An alphanumeric string of up to 7 characters that identifies where the rack is physically located. (O)

Engineer - Enter the name of the engineer in charge of the rack. (O)

Description - A description of the rack; an alphanumeric string of up to 40 characters. (O)

Step 3. Select OK to add the rack to the database or Cancel to return to the System Configurator without adding the rack.

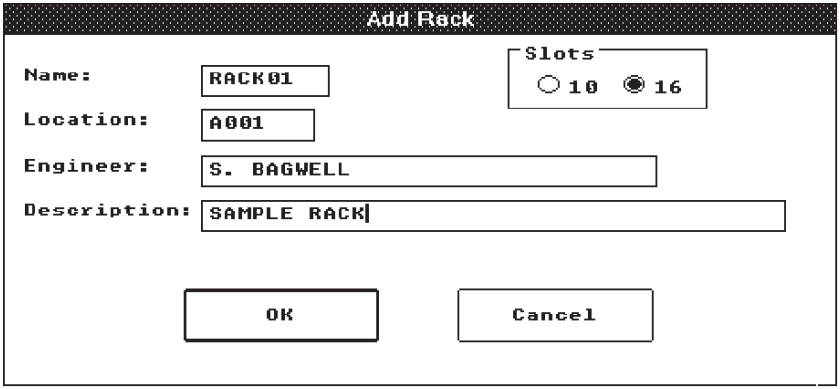


Figure 5.10 - Add Rack

5.3.2 Modifying Rack Information

Select Modify Info to display the Modify Rack dialog box for the selected rack. You can make changes to any information except the rack name. If there is an error in the name of a rack, you must remove the rack, then re-enter it using Add. Select OK to save the changes to the database or Cancel to return to the System Configurator without modifying the rack information.

5.3.3 Removing a Rack

Select Remove to delete one or more selected racks. A confirmation box will appear on the screen. Note carefully that the selected rack(s) and its modules, tasks, and variables will be deleted from the database if you confirm the command using OK.

5.3.4 Displaying Rack Information

Select Display Info to display the Display Rack dialog box, which lists the rack name, location, engineer, description, and number of slots. You cannot make changes to any of the information displayed. Select OK to return to the System Configurator.

5.3.5 Editing the Documentation File for a Rack

Select Document from the Rack menu to access the text editor, which you can use to create or edit the documentation file associated with the selected rack. The file will be named automatically (the rack name with an .RDC extension) and will be stored in the system subdirectory under the library directory. See section 4.21 for instructions on using the text editor.

5.3.6 Generating Rack Cross Reference Files

For each common variable, the rack cross reference lists the variable's type, hardware address, description, and the tasks that use it. The rack cross reference then lists errors or warnings for the rack, such as unconfigured variables, variables written to more than one task, and unused variables. Table 5.1 provides a complete list of all the errors and warnings that can be generated with the rack cross reference.

You can generate a rack cross reference for one or more racks in a system. Select one or more racks from the Rack list. Then select Generate Rack Cross-reference from the Rack menu. The rack cross reference files will be created for the selected racks.

The fields described below are listed across the top of the rack cross reference printout. See figure 5.11 for an illustration of the rack cross reference.

Library:I:\AMXLIB System:UDCTEST Section:ANET Rack:A00 Task/File: 08-Nov-93 11:00:58 Page: Rack Xrf – 1
Generation Date: 08-Nov-93 11:01:19

Variable Name	Type	S	D	RS	REG	B	Description	Task Cross Reference
AHI%(4095)	Memory				16		ARM VOLTS AT CEMF HIGH READING	S1EMF* S1PEMF*
AI_LOAD_REF%	I/O	4			7		test drive load reference from pot	S1REF2 S4ASPD2
AI_MOD_LOAD%	I/O	4			8		test drive modulate load signal from pot	S1REF2 S4ASPD2
AI_SINE_WAVE%	I/O	4			9		test drive sine wave ref excitation	← UNUSED→
AI_SPD_REF%	I/O	4			6		master speed reference from pot	S1REF2
ALO%(4095)	Memory				17		ARM VOLTS AT CEMF LO READING	S1EMF* S1PEMF*
COLLECT	HW_Event							S1EMF
COM_PC_D01_FLT@	Memory				60		DROP 1 COMM. FAULT LATCH	S1UDC4A*
D01_OK@	I/O	6			4	1	DROP 1 COMMUNICATION OK BIT	S1UDC4A
NETW_OK@	Network	6	0		32	0	NETWROK OK BIT FOR A00	S1UDC4A* S1UDC5A
NET_CUR_RATE%	Net-Wide	6	0		35		current loop reg. ramp rate to drop 1	S1REF2*

Legend:

S = Slot; D = Drop; RS = Remote Slot; REG = Register; B = Bit
(The register listed for memory variables is the items is the item number in the Software Configurator form.)

An address containing “?” means that the network-wide name for this variable was not found in the Network database. Add the network-wide name or use the “Repair” command in the Rack Configurator to remove unresolved variables.

* = Variable modified in this task

Figure 5.11 - Rack Cross Reference

VARIABLE NAME - the variable name including type and array dimensions

TYPE - the variable type:

I/O - mapped to a module in the master rack other than an Interface or Network module or to set-up registers on an Interface or Network module

Rem I/O - mapped to a Remote I/O module or Rail

AutoMate - mapped to an AutoMate Interface module

AB Link - mapped to an A-B Interface module

Modbus - mapped to a Modbus Interface module

Network - mapped to a Network module without using a net-wide name

Net-wide - mapped to a Network module using a net-wide name

Memory - mapped to common memory module (volatile)

nvMemory - mapped to common memory (non-volatile)

Reserved - variable has the same name as a reserved name (e.g., BATTERYSTATUS@)

Unknown - used in a task but not mapped to an I/O or common memory location

HW_Event - hardware event

SW_Event - software event

S - the master slot number

D - the remote I/O drop number (for remote I/O); the drop area number (for Network); B0, B1, N0, N1 (for A-B Interface); [blank] (otherwise)

RS - the remote slot number (for remote Shark and Multibus I/O); Remote Head port (for AutoMate I/O); [blank] (otherwise)

REG - Local Head port (for AutoMate I/O); the register in octal (for AutoMate Interface); the register in decimal (otherwise)

B - the bit number for booleans, [blank] otherwise

DESCRIPTION - the variable description (from network-wide database for linked network variables; description in the variable database if present; otherwise, description from the first task that has one; otherwise blank)

TASK CROSS REFERENCE - names of the task(s) which use the variable; no extension; '*' if variable is modified

Table 5.1 - Rack Cross Reference Warning and Error Reports

Warning or Error Listed (W or E)	Item listed
Total number of variables (E)	number
Tasks defined in slots with no Processor or UDC (E)	tasks
Tasks with serious compilation errors (no cross reference information available) (E)	tasks
Tasks with minor compilation errors (potentially incomplete cross reference information) (E)	tasks
Tasks with undefined variables (variables used in the task with no LOCAL or COMMON definition) (E)	tasks
More than 32 events found in tasks (E)	
More than 100 unconfigured variables found in tasks (E)	
Unconfigured variables (variables declared as COMMON in tasks but not configured) (E)	variables with tasks which use the variable
Unresolved network variables (variables with network-wide name not in the network database) (E)	local and network-wide variables
Variables with array dimension mismatches (variables with different array dimensions between the task's definition and the database) (E)	variables with tasks which use the variable (do not include array dimensions)
Variables with string length mismatches (variables with different string length between the task's definition and the database) (E)	variables with tasks which use the variable (do not include array dimensions)
Variables with duplicate definitions (variables mapped to more than one I/O location).	variables
Variables configured to the same I/O locations (a pair of variables mapped to the same I/O location) (E)	pairs of variables
Variables which modify the same I/O location (E)	pairs of variables
Memory variables which are read but never written (E)	variables with tasks which use the variable
Variables which should not be configured since they conflict with events or reserved names (E)	variables
Predefined variables used but not configured as memory variables (e.g. BATTERYSTATUS0@) (E)	variables
Variables written to UDC register pairs in which a UDC task writes a bit (variables which are: (1) written by an AutoMax task, (2) mapped to a UDC register; (3) within a register pair on an even boundary (300 and 301, 302 and 303, not 301 and 302) where a UDC task writes any bit within the register pair (E)	variables with tasks which modify the variable
Variables used in a UDC task but not configured on the UDC modules (E)	variables

Table 5.1 - Rack Cross Reference Warning and Error Reports (Continued)

Unused variables which are configured but not used in any task (W)	variables
Variables modified in more than one task (W)	variables with tasks which modify the variable
Volatile memory variables which are used in only one task (W)	variables with tasks which use the variable
Memory variables which are written but never read (W)	variables with tasks which use the variable

5.3.7 Printing Rack Files

The Print Rack command enables you to print out files relevant to one or more racks. After selecting one or more racks from the Rack list, select Print from the Rack menu. A dialog box with the various print options will be displayed (see the figure below).

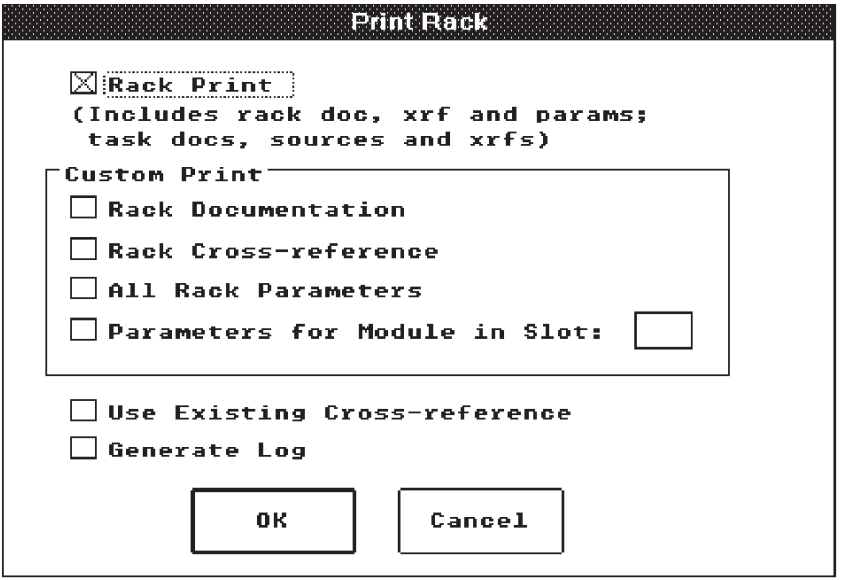


Figure 5.12 - Print Rack Dialog Box

Selecting **Rack Print** will print out the following for each of the racks selected:

- Rack title page (rack name, description, engineer, location, and a list of the tasks in the rack).
- Rack documentation file
- Rack cross reference file
- Rack parameter file (only for racks containing UDC modules)
  - Task title page (task name, description, engineer, slot, priority, critical, reconstructible, utility).

- Task documentation file
- Task source code
- Task cross reference file

If you de-select (uncheck) Rack Print, the Custom Print options will be enabled.

Select **Rack Documentation** to print out the rack documentation file.

Select **Rack Cross Reference** to print the cross reference file for the selected rack(s).

Select **All Rack Parameters** to print the drive parameters for all the UDC modules in the rack.

Select **Parameters for Module in Slot:** to print the drive parameters for a particular UDC module in the rack. You must enter the slot number of the UDC module.

Note that the two rack parameters options are mutually exclusive.

Select **Use Existing Cross Reference Files** if you know that the cross reference files are current. Selecting this option will save time since new files will not be generated before printing. If cross reference files do not already exist, they will be generated and printed. Note that cross reference files generated by the print routine will be deleted after printing.

Select **Generate Log** to create a log file that will list any errors that may occur during the print job. Each rack will have a separate log file. The log file will exist under the rack directory and will be named _PRINT.LOG. If there are no errors, the log file will simply list the files that were printed.

5.3.8 Transferring a Rack

You can copy all the files for one or more selected AutoMax Executive racks to diskettes (or another hard drive, or another library on the same drive), or copy all the files for an AutoMax Executive rack from diskettes (or a hard drive) into the current AutoMax library (see section 5.1.7 for additional information). If you are transferring to diskettes, you will need at least one diskette for each rack. You will be prompted for additional diskettes, if needed. Diskettes can be formatted as needed or, to save time, you can use blank pre-formatted diskettes. Use the following procedure to transfer racks. Figure 5.13 illustrates the Transfer Rack dialog box.

Step 1. Select the rack(s) to be transferred from the rack list.

Step 2. Select Transfer from the Rack menu. The Transfer dialog box will be displayed with the current drive and library (as designated in AutoMax Setup) listed in a box on the top left.

Step 3. Enter information for the following fields. Some fields may contain default entries. If you do not change these fields, the default will be used.

Direction arrows - Transfer the rack To (left arrow) or From (right arrow) the current library. Select the appropriate option button.

Drive - Enter the source drive or the destination drive. If A or B is entered, the Executive software assumes the drive

is a floppy drive; otherwise, AutoMax assumes a hard drive.

Library - Enter the library directory to copy the rack(s) from or to.

Format Floppies - Select if you want to format destination diskettes.

High Capacity - Select if you want to format and/or write to high density diskettes. The appropriate diskette capacity will be displayed.

Rack(s) - The Executive software will list the rack(s) selected when the Transfer command was called. If you are transferring out of the current library, make certain that this field lists all of the racks you want to transfer. If you are transferring racks in, enter the rack names separated by commas or spaces.

Step 4. Select which of the following groups of files you want to transfer with each rack:

Object files - application task object files, parameter object files, and rack configuration object file(s).

Cross-reference files

Transfer tokens

Step 5. Select Help for information relevant to the Transfer procedure, Cancel to return to the System Configurator without transferring a rack, or OK to begin the rack transfer.

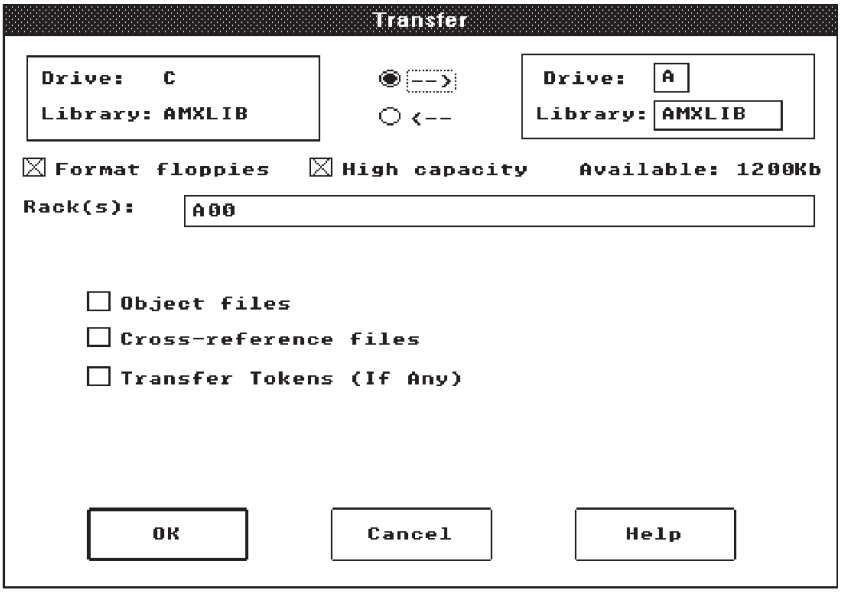


Figure 5.13 - Transfer Rack

5.3.9 Importing a Rack from AutoMax Version 1 or Version 2

You can import the configuration information and application tasks for an AutoMax Version 1 or AutoMax Version 2 rack and any Remote I/O racks into AutoMax Executive V3.x. The rack name must not already exist in the selected system. **If you want to use tasks created with earlier versions of the AutoMax V3 Executive in a V3.5 environment, see Appendix Q.**

Before importing an AutoMax V1 rack, you must edit any Control Block tasks as follows: decimal points must be added to any real literal values that do not already have them. Blocks that allow real literal values include the following: DIFF_LAG, INTEGRATE, LAG, LEAD_LAG, PROP_INT, and PID. After adding the decimal points where appropriate, the task must be re-compiled. No changes are required for tasks in AutoMax V2 racks.

It is helpful to try to create a test rack in V3.x first to understand the structure by which the configuration task and application tasks will be organized when they are imported into V3.x. The Import command will not delete the original files or alter them in any way.

If your system has .NET files, you must run NETIN to compile the .NET files and generate a network-wide database prior to importing AutoMax V2 racks. Refer to Appendix G for the NETIN procedure.

The Import procedure will add the variables and their descriptions from the AutoMax configuration task to the rack's variable database. The task names, types, priorities, critical flags, and Processor slots from the configuration task TASK statements are added to the task database.

The Import procedure will copy in the task source files for the rack. Note that tasks must be compilable in AutoMax V2 in order to import properly. Since BASIC include (.INC) files are treated as tasks in the current Executive software, a task is added for each include file found.

5.3.9.1 Module Types Recognized by the Rack Import Procedure

Import will add as many modules as possible from the variable definition statements in the configuration task. The following list shows the modules added for the type of definition found.

Definition Found	Type of Module Added
NETDEF or GBLDEF	57C404 Network Communications
MODDEF	57C414 Modbus interface
RIODEF	57C416 Remote I/O Communications (in the master rack)
RNETDEF	57C417 AutoMate interface
ABDEF	57C418 AB interface
TASK	57C430 AutoMax Processor
IODEF	Generic module (unless one of the other definitions is found for the slot)

5.3.9.2 Generic Modules

Note that a “generic” module is added to the appropriate slot in the local rack when IODEF statements are found for the slot in the configuration task. The generic module is used because the IODEF statement does not provide enough information to determine the specific type of module used. Note also that for a remote I/O network, no hardware is added to the rack database for the same reason.

5.3.9.3 Accessibility of Variables Added by Import

Although the Remote I/O hardware which was configured in the AutoMax V2 configuration task does not get added to the rack, all variables defined for it are added to the rack database. In the local (master) rack, a generic module will have the variables defined for the corresponding slot mapped to it. Remote I/O variables are not accessible to the user until the correct module or rail is actually added to the Remote rack or Head through the Rack Configurator.

Variables that exist in the rack database, but are not attached to any module at all, are called floating variables. These variables must be either deleted from the databases or the correct modules added to account for them to properly document your application. Note carefully that when you configure the remote I/O network hardware in the Rack Configurator after the import procedure, you must be careful to add the correct hardware (proper size rack or Remote Head) the first time. This is important because if you make a mistake and add the wrong hardware, the only method of modifying the configuration is to remove the rack or Head and add the correct rack or Head. However, removing the rack or Head will also delete all of the floating variables that are mapped to the remote I/O system. When you configure the remote I/O system, you will be prompted that variables have already been configured for the hardware.

If floating variables exist in a remote I/O network when the configuration object file is generated in the Task Manager, they will be included in the configuration object code even though they are not accessible in the Variable Configurator. Floating variables in the master rack are flagged as errors. You can compile tasks when there are floating variables in a remote I/O network. You cannot, however, compile tasks when there are floating variables in a master rack.

Local network variables will be linked to network-wide names whenever possible. If there is a network-wide name for the network point used by a variable, the network-wide variable is linked to that variable. Any network or remote I/O variables configured as both single and double integers will not import correctly. The procedure will result in prompts that indicate “duplicate Multibus addresses” and will require you to delete the offending variables.

5.3.9.4 Register Assignments on Modules Added by Import

Registers defined for some modules in AutoMax V2 cannot be configured in the current Executive software and will cause an error message to be displayed (invalid/incompatible data type) when you try to configure the module if you import the rack without first checking your AutoMax V2 configuration. See section 5.1.8.4 for more information. Refer to Appendix K for examples.

5.3.9.5 Source Drive and Subdirectory for Import

It is important that the files to be imported be stored in a specific manner as follows. The only configuration task in the subdirectory should be the configuration task for the rack you want to import. The configuration task does not need to be compiled because the import procedure will compile it. If there is more than one configuration task in the subdirectory, the import procedure will use the first one found, which may not necessarily be the configuration you want to import.

The Import procedure will copy in all the application tasks (.BAS, .BLK, .PC. and .INC) found in the subdirectory. However, only those tasks with corresponding TASK statements in the configuration task will be added to the tasks database. If no application task exists for a TASK statement, you will be prompted to delete the task when you enter the Task Manager. Utility tasks (those without TASK statements) can subsequently be added in the Task Manager, which can use the imported source file of the same name.

5.3.9.6 Procedure for Importing a Rack

Use the following procedure to import a rack.

Step 1. Select Import from the Rack menu. Note that this assumes you have selected a system and section first. The Import V2 Rack dialog box will be displayed.

Step 2. Enter information for the following fields, then select OK. Some fields may contain default entries. If you do not change these fields, the default will be used.

Rack name - The name you want to use for the rack in the currently selected; an alphanumeric string (unique within the system) of up to 8 characters (no spaces, hyphens, or underlines). The name of the rack in the new system may be different than the name of the rack being imported, i.e., the subdirectory in which the rack is located.

10-slot or 16-slot rack - Select the appropriate type of rack.

Location - An alphanumeric string of up to 7 characters that identifies where the rack is physically located. This is not the drive\directory location of the rack.

Engineer - Enter the name of the engineer in charge of the rack.

Description - A description of the rack; an alphanumeric string of up to 40 characters.

Step 3. A dialog box for the Source Directory will be displayed on the screen. Enter the rack source directory path, including the drive and subdirectory. Type over the default path displayed if changes are required.

Step 4. Select OK to start the import or select Cancel to return to the Rack Configurator without importing the rack.

If the import process is unsuccessful, the rack created at the beginning of the procedure will be deleted from AutoMax Executive V3.x automatically.

5.3.10 Selecting All Racks

The Select All command selects all the racks listed for the selected system and section. Select All can be used when you want to transfer, print documentation files, or remove all of the racks for a selected section. To select one rack after selecting all, simply click on the desired rack.

5.3.11 Configuring Racks

Select Configure Rack from the Rack menu (or double-click the left mouse button on the desired rack) to access the Rack Configurator, which is used to add modules to a rack, display or modify information for existing modules, move modules to other slots in a rack, or remove modules from a rack. You can also access the Variable Configurator from the Rack Configurator. See section 6 for how to use the Rack Configurator.

5.3.12 Creating Application Tasks

Select Manage Tasks from the Rack menu to access the Task Manager. The Task Manager is used to edit, copy, transfer, compile, and print tasks, as well as to add the tasks to the selected rack. The On line menu, which allows you to download application tasks to the AutoMax rack, put them into run, and monitor, is also available from the Task Manager. See section 8 for how to use the Task Manager.

5.3.13 Using System Tokens and Token Passwords

If you are using tokens to protect your database files, the command “Set Token Password” will appear on the Rack menu. Refer to Appendix N for a description of how to use this command.

5.3.14 Using Version Control Library

If your AutoMax system is using the Version Control Library facility, the command “Version Control Library” will appear on the Rack menu. Refer to Appendix P for descriptions of how to use the VCL commands.

5.4 Commands Menu

The Commands menu on the System Configurator screen is used to edit DOS files that were not created by AutoMax Executive V3.x, to access the Kermit communications utility, and to load the operating system to the Processor(s) and UDC modules in the rack. Note that the operating system for the Processors in the rack must be loaded before any communication can take place with the rack.

5.4.1 Editing DOS Files

Select Edit Any File from the Commands menu to edit DOS files only. Except for .LOG and .XRF files, you cannot use this command to edit files created by the Executive software. Figure 5.14 illustrates the Edit a File dialog box.

A list box with all the available directories and files in the current drive\library will be displayed. You can change the path displayed by entering the desired path in the space provided. Select the file you want to edit. Although all files in the subdirectory will be listed, they cannot all be edited. The Programing Executive will display a message if the selected item cannot be edited. You can make the selection from the list box or you can enter the filename in the text box. The filename can contain a wildcard character (*). All matching files will be displayed in the list box. For example, entering *.TXT would display all the text files in the current drive\directory and allow you to select the file to be edited. After you make a selection and click on OK, the text editor will come up on the screen.

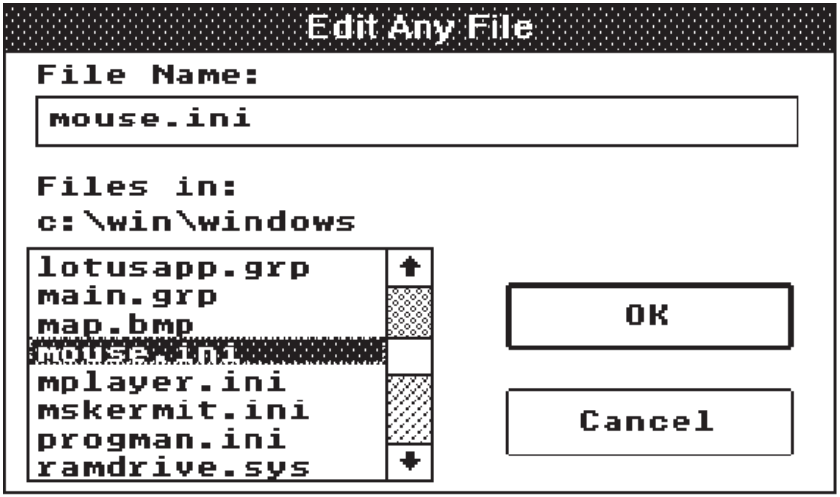


Figure 5.14 - Edit a File

5.4.2 Kermit Communications Software

Select Kermit from the Commands menu to access the Kermit communications utility. Kermit allows your computer to emulate a terminal or to transfer software to and from another computer. Your computer must first be connected to the host computer or another personal computer via a serial port. See the Kermit Reference Manual (J-3616) for information about using Kermit. Type EXIT at the KERMIT prompt to close the Kermit utility and return to the System Configuration screen.

5.4.3 Loading Operating Systems

AutoMax Processor modules and UDC modules require operating systems to be loaded before they can function in the rack. The AutoMax Programming Executive software includes the operating systems for the AutoMax Processors. **Operating systems for the Universal Drive Controller (UDC) modules are included only on the separately-purchased DPS drive software. This software is assumed to be loaded onto the personal computer if it is required.** The UDC operating system file also contains the operating systems for the Power Module Interface (PMI) Processors. Once the UDC operating system has been loaded to the UDC module, the PMI operating system is automatically loaded to the PMI Processor(s) upon power-up.

Before you can go online with any rack in the system, the operating system for the AutoMax Processor(s) must be loaded to the local rack from the personal computer on which you have installed the AutoMax Programming Executive software. If you are running Version 3.5 of the Programming Executive software, and the local rack already contains Version 2.0 or later of the AutoMax operating system, you can communicate with any rack that is on the same network as this local rack. However, you will be able to use only those features that are supported by the operating system which exists on the destination rack. For example, before you can load the UDC operating system to the UDC module(s) in a rack, Version 3.6 of the AutoMax operating system must be loaded to the AutoMax Processor(s) in that rack. If you are running an earlier version of the Programming Executive software, you will not be able to communicate with a rack that contains the AutoMax Version 3.6 operating system either by direct connection or over the network.

The leftmost AutoMax Processor in the rack will check for compatibility between the AutoMax OS and the UDC OS. If you replace a UDC with another UDC that already contains an OS that 's incompatible, the new UDC will be disabled and its "OS OK" LED will turn off.

Precautions in Loading the AutoMax Operating System

Normally, when the operating system is loaded to the AutoMax Processors in a rack, the AutoMax operating system currently on the Processors (if any) is written over. However, it is possible that the operating system in the rack is a later version than the operating system contained in the Programming Executive software you are using. In this case, the Programming Executive will be unable to establish communication with the rack in order to load the new AutoMax operating system until you erase the existing operating system. The procedure for erasing the operating system for all of the AutoMax Processors in a rack is described in section 5.4.4.

The situation described above cannot occur when loading the operating system to UDC modules since communication is established first with the AutoMax Processor in the rack first and then with the UDC module rather than the UDC module directly. The existing operating system on a UDC module (if any) will be written over when you load any UDC operating system to that module.

AutoMax Processor modules have on-board battery backup, so you will need to re-load the AutoMax operating system only when enhancements become available, when you change the password for the rack using the PWOS.EXE utility described in 9.2.1, or when an AutoMax Processor module in the rack is replaced. Note carefully

that **all tasks in the rack will be deleted** when you load the AutoMax operating system to the Processor(s) in the rack. **To ensure that the values of any tunable variables will be maintained in their respective tasks, you should save all application tasks in the rack that contain tunable variables back to the personal computer from the rack (see 14.4) after you have finished tuning variables, and before you load the new operating system.** If you are loading a more recent version of the operating system, you may need to re-compile your application tasks using the corresponding Programming Executive software before downloading them to the rack.

UDC modules use non-volatile Flash memory to store their operating systems and UDC application tasks, so you will need to re-load the UDC operating system only when enhancements become available or when a UDC module in the rack is replaced. Note carefully that **all tasks in the UDC module will be deleted** when you load the UDC operating system to that particular UDC module.

See Appendix T for problems that can occur when you load operating systems.

Types of AutoMax Operating Systems

The AutoMax operating system, which oversees the operation of the AutoMax Processors and the execution of application tasks, is provided in three versions: 6010/6011-Standard, 6010/6011-Ethernet, and 7010-Standard. The 6010/6011-Standard or 6010/6011- Ethernet operating system can be loaded onto M/N 57C430A or 57C431 Processors. In order to use the Ethernet functions that allow communication over Ethernet using the TCP/IP protocol, an Ethernet Network Interface module (M/N 47C440) must be installed in the rack and the Ethernet operating system must be used. The Ethernet operating system is also required if any of the following functions are used in BASIC tasks: READVAR%, WRITEVAR%, FINDVAR!, and CONVERT%. The 7010 - Standard operating system, which supports all of the Ethernet functions, must be used with the M/N 57C435 Processor.

When you load the AutoMax operating system to the Processor modules in the rack, you will be prompted for which operating system you want to load for the M/N 57C430A and M/N 57C431 Processors. The 7010-Standard operating system will be loaded to all M/N57C435 Processors. Note that if you have loaded the 6010/6011-Standard operating system and then attempt to use the Ethernet communication functions in an application task, the Processor will display error code 4A on its LEDs when you try to put the task into run.

The 6010/6011-Ethernet operating system will occupy approximately 134K of RAM, leaving 122K available for application tasks on the M/N 57C430A Processor. The 6010/6011-Standard operating system will occupy approximately 119K of RAM, leaving 137K available for application tasks on the M/N 57C430A Processor. The M/N 57C431 makes 300K available for application tasks, regardless of which operating system is used. The M/N 57C435 also makes 300K available for application tasks.

Time Required to Load Operating Systems

The operating system(s) will be loaded at the maximum baud rate available for the Processor being used. If you are using 6010/6011 and 7010 Processors in the same rack, make sure that the leftmost Processor is a 7010 (M/N 57C435) Processor. This will allow the

operating system(s) to be loaded at 19200 baud. If the leftmost Processor in the rack is a M/N 57C430A or M/N 57C431, the operating system(s) will be loaded at 9600 baud. At 9600 baud, it will require approximately two minutes to load each operating system to the Processors in the rack. If there are 6010/6011 and 7010 Processors in the same rack, it will take approximately twice as long as it would if there were only one type of Processor in the rack. The UDC operating system requires approximately four minutes to load to each UDC module in the rack.

What to Do Next

After loading the appropriate operating systems to the rack following the instructions below, you would normally load compiled application tasks and (optionally) UDC drive parameters to the rack. See section 14.3 for more information.

5.4.3.1 Procedure for Loading All Operating Systems to the Rack

The flowchart in figure 5.15 illustrates the options you have when loading the AutoMax operating systems and UDC operating systems (if DPS drive software has been installed). Before loading the operating system(s), review this flowchart carefully.

Use the steps that follow to initiate the Load Operating Systems procedure.

Step 1. If you have not already done so, turn on the personal computer and run the AutoMax Programming Executive by typing

AUTOMAX3

Step 2. Turn on power to the rack. Note that if the AutoMax Processors in the rack do not have an operating system, the leftmost AutoMax Processor module in the rack will display the letters “L” and “O” (reading top to bottom). This code prompts you to load the operating system. Because Processor modules have onboard battery backup, you will need to load the operating system only when enhancements become available, when you change the password for the rack using the PWOS.EXE utility described in 9.2.1, or when a Processor module in the rack is replaced.

Step 3. Connect the personal computer to the leftmost AutoMax Processor in the rack, following the directions in section 3.4.

Step 4. Select the system for which you want to load operating system(s).

Step 5. Select **Load OS Single Rack** from the Commands menu. You will see the following message.

> Connect cable to processor. Depress ENTER key when ready, ESC key to abort.

Step 6. Press ENTER.

The Programming Executive will establish communication with the local rack and will determine the type(s) of AutoMax Processors in the rack. It will also determine whether operating systems are currently loaded to the Processors in the rack.

The prompts displayed on the screen will depend on your system configuration and the options you choose as you progress through the Load Operating System procedure.

Note: *If you are loading an operating system to a Processor or UDC that already contains one, you will be required to enter the rack password before the operating system can be loaded.* ***Loading the operating system to an AutoMax Processor or UDC module that already has one will cause all application tasks in the rack or the UDC module to be stopped and deleted.***

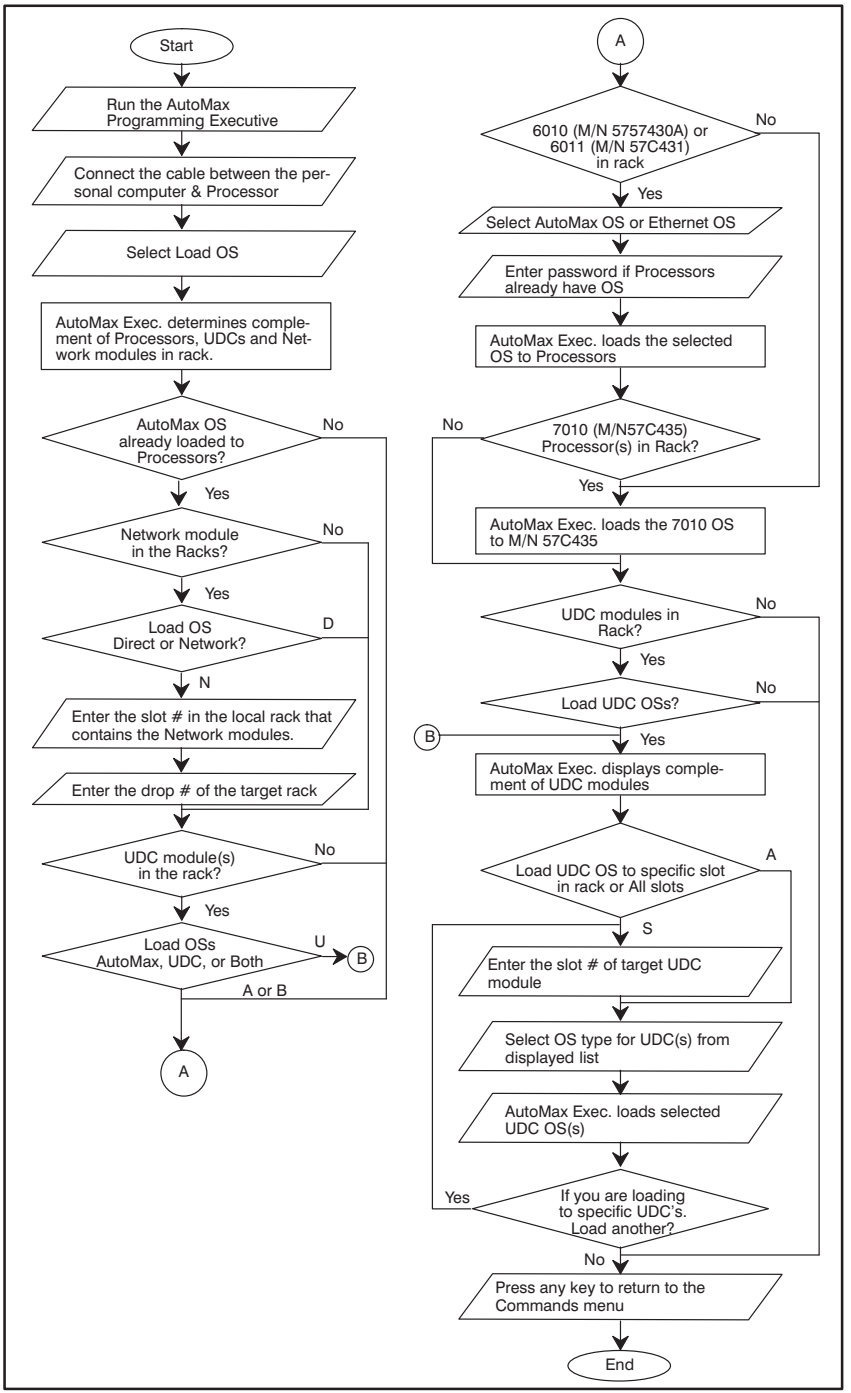


Figure 5.15 - Load Operating System

5.4.3.2 Loading the AutoMax Operating System to All Drops on One or All Networks

The procedure that follows describes how to load the AutoMax operating system to the AutoMax Processors in all racks on a single network or all networks to which the local rack is connected.

Step 1. If you have not already done so, turn on the personal computer and run the AutoMax Programming Executive by typing

AUTOMAX3

Step 2. Turn on power to the rack.

Step 3. Connect the personal computer to the leftmost AutoMax Processor in the rack, following the directions in 3.4.

Step 4. Select the system for which you want to load operating system(s).

Step 5. Load the AutoMax operating system to the local rack using the procedure described in 5.4.3.1, and then return to the main screen (System Configurator).

Step 6. Select **Load OS Network Wide** from the Commands menu. You will see the following message.

Connect cable to processor. Depress ENTER key when ready, ESC key to abort.

Step 7. Press ENTER.

The Programming Executive will establish communication with the local rack and will determine the types of AutoMax Processors in the rack and that there are one or more Network modules in the rack. The following message will be displayed on the screen.

Enter the slot number of the network card to go out on, or A to load the OS to ALL drops on ALL networks where there is none, or O to do the same but also write OVER any existing OS.

Network card found in slot(s):
xx xx

All - To load OS into ALL drops on ALL networks where there is no OS.

Overwrite - Same as above but also write over any existing OS.

Network Card Slot Number (or Option):

Step 8. Enter the slot number of a network module in the local rack to load the operating system to Processors on all drops on a single network;

or

Enter “A” to load the operating system to Processors that do not have an OS on all drops on all networks;

or

Enter “O” to load the operating system to Processors and overwrite any existing operating systems on all drops on all networks.

If you entered a network module slot number, you have the same options as above: select (A) to load the operating system to all Processors that do not have an operating system on all drops on the network, or select (O) to overwrite any existing operating systems as well as loading the operating system to the Processors on all drops on the network.

If a rack contains only M/N 57C435 (7010) Processors, the 701x operating system will be loaded to all the AutoMax Processors in the rack. The AutoMax Executive will display on the screen the portion (%) of the operating system that has been loaded. Go to step 10.

If a rack contains any M/N 57C430A (6010) or 57C431 (6011) Processors, the following message will be displayed.

You may either load the AutoMax operating system or the AutoMax operating system with the Ethernet option.

Enter A for AutoMax or E for AutoMax with Ethernet (A)utomax or (E)thernet [A]:

Step 9. Enter “A” or “E”. The selected operating system (601x or 601xe) will be loaded to all the 6010 and 6011 AutoMax Processors in the rack. The 701x operating system will be loaded to all of the 7010 Processors in the rack. The AutoMax Executive will display on the screen the portion (%) of the operating system that has been loaded.

Step 10. If an error occurs during the process of loading the operating system(s), an error message will be displayed on the screen. Repeat steps 8 and 9 above.

5.4.4 Erasing Operating Systems

Select Erase OS from the Commands menu to erase the operating system(s) from AutoMax Processors or UDC modules in a rack (return the Processor or UDC to boot). Note that when AutoMax operating systems are deleted, the rack configuration file will be deleted and all AutoMax tasks in the rack will be stopped and deleted; when the operating system for a UDC module is deleted, its parameter object file (.POB) will be deleted and its tasks will be stopped and deleted.

This procedure and screen prompts are similar to those used to load the AutoMax and UDC operating systems to the rack, but instead of loading an OS, this procedure erases the OS that currently exists on the Processors or UDC modules you select.

If the personal computer is connected using a direct serial connection and the left-most Processor contains an operating system, you will be prompted to select the destination rack (over the network or direct) just as is currently done for loading the operating system.

If the personal computer is connected using the PC Link card, you will be prompted to select the destination rack (over the network or direct) just as is currently done for loading the operating system.

If any UDC modules exist in the destination rack, you will be prompted for which operating systems to erase (AutoMax or UDC). The warning “Erase UDC OS before AutoMax OS when erasing all Processors” will be displayed with the prompt. You can choose to erase either the OSs on all Processors in the rack or the OS for a UDC module in the slot you designate. If you want to delete the operating systems for all AutoMax Processors and UDC modules in the rack, you must delete the UDC OSs first.

5.4.5 Repairing Databases

In some cases, the Programming Executive will display an error message which states that the database index files do not match the database files. This should not occur during normal operation. Usually, this error occurs if the database files have been changed without using the Programming Executive or if a Programming Executive operation was interrupted by turning off the personal computer. This message will instruct you to use the **Repair System Databases** or **Repair Rack Databases** command. These commands will re-index all index files for the selected system or selected rack, respectively.

5.5 Setup Menu

The Setup menu is used to modify the Windows configuration and to set up AutoMax Executive V3.x. You should set up both AutoMax Executive V3.x and Windows when you install the software.

5.5.1 Modifying the Windows Configuration

Select Windows from the Setup menu to make changes to the Windows configuration. The Windows Control Panel will be displayed. See the *Microsoft Windows User’s Guide* for instructions on using the Windows Control Panel.

Note that in order to print PC tasks (ladder diagrams) and cross reference files properly, your printer must be capable of printing in compressed mode (132 columns). You can choose to print in either landscape or portrait orientation. All printed output will be generated according to the parameters defined in the Printers setup in Windows.

5.5.2 Modifying the AutoMax Executive Configuration with Setup

Select AutoMax from the Setup menu to make changes to the AutoMax Executive configuration. The Setup dialog box (see figure 5.16) will display default entries. You can make changes to any the following fields:

Library Drive - Enter the letter of the drive containing the library you want to work with.

Library Directory - The library is the base directory which will contain the systems you want to work with. The default library is AMXLIB. You can accept the default or enter the name of your library directory.

Version Control Library Drive - Enter the letter of the drive you want to use for VCL files. (Optional)

Version Control Library Directory - Enter the name of the directory that will contain your VCL files. (Optional)

Note: *You must check the PVCS Installed checkbox before you can access the Version Control Library fields.*

Text Editor - The Norton editor is the default text editor. The text editor is used to edit documentation files, BASIC tasks, Control Block tasks, include (.INC) tasks. It is also called up when you select Edit File from the Commands menu. If you want to use a different text editor, enter the filename of the text editor (including the .COM or .EXE extension) you want to use with AutoMax. See section 4.21 for additional requirements. Note carefully that you cannot delete the Norton editor because it is required for creating Ladder Logic language REMARK sequences.

Drive for temporary files - Enter the letter of the drive to contain temporary files. A non-network hard disk drive is required for temporary workspace to ensure exclusive access. (R)

Switch for formatting a 5-1/4 floppy to 360Kb: Enter any DOS formatting parameters required to format a $5^1/_4$" floppy to 360Kb for your floppy disk drive. For example, if you wanted to format a low density disk in high density drive A, the DOS format command would be FORMAT /4 A:, so you would enter /4 for this field. (O)

Switches for formatting a 3-1/2 floppy to 720Kb - Enter any DOS formatting parameters required to format a $3^1/_2$" floppy to 720Kb for your floppy disk drive. (O)

PVCS Installed - Select this check box if you have PVCS on your network drives and you want to use the VCL facility. See Appendix P for more information.

User Name - Enter the user name using up to 31 characters. The user name will appear in system messages.

Exit Windows when exiting the AutoMax Programming Executive - The default is to exit Windows when you exit the AutoMax Programming Executive. If you de-select this check box, you will return to the Windows Program Manager when exiting AutoMax.

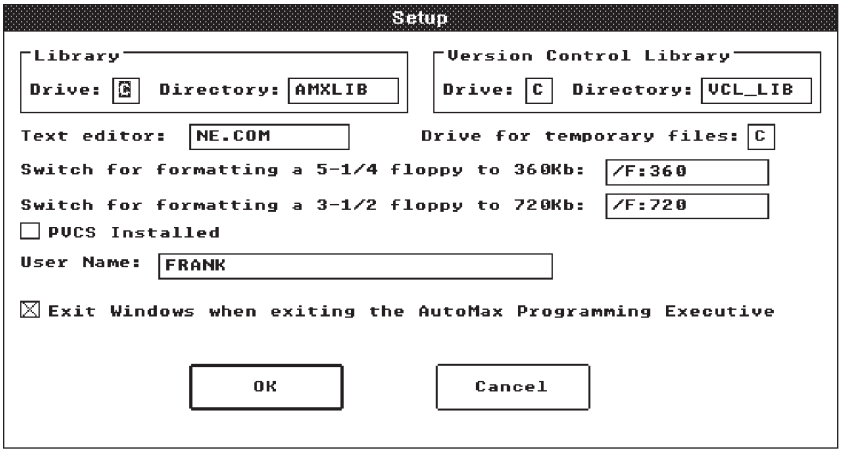


Figure 5.16 - AutoMax V3 Setup

When you have finished making changes, select OK, or Cancel if you have changed your mind.

5.5.3 Modifying the Online Communication Configuration

Select Communication from the Setup menu to choose the method you want to use to communicate with racks. The Communication Setup dialog box will be displayed. You can choose either communication via the serial port on the AutoMax Processor or communication via the PC Link module.

If you want to use a serial port for communication (the default), you must specify the port number (1 [default] or 2) and the baud rate (1200, 2400, 4800, 9600, 19200 [default]).

If you want to use a PC Link module for communication, you must specify the drop number (1-55), the base memory address (D000 [default], D400, D800, DC00), and the port address (250H [default]) for the PC Link module. These parameters must match the parameters set when you installed the PC Link module in the personal computer (see J2-3011). The PC Link module will be configured with a drop depth of 1. If there is more than one PC Link module in the personal computer, AutoMax will use the one whose base memory address and port address match the addresses entered in the Communication setup.

5.5.4 Modifying the Print Configuration

Select Print from the Setup menu to make changes to the printer configuration. The Print Setup dialog box will be displayed. You can make changes to the following fields. If no changes are made, the default settings will be used. Note that duplex printing (selected on the Printers Configuration screen using the Windows Control Panel) is supported.

3-Hole Punched Paper - This option is useful only when printing in landscape mode. Check this box if you want to leave an extra four lines (1/2 inch) at the top of the page for 3-hole punched paper. These four lines will be included in the number of lines per page, so there will be four fewer lines of print on each page. The default is unchecked (not selected).

Lines Per Page - Select Full Page or Lines Per Page.

If you select Full Page, AutoMax will automatically print the maximum number of lines that can be printed per page based on the information entered on the Printers configuration screen in the Windows Control Panel. This number will be displayed next to the Full Page option.

If you select Lines Per Page, enter a number between 20 and one less than the number displayed for Full Page.

5.6 On_Line! Command: Accessing Online Functions

Select On-Line! to access the online functions. These functions are described in sections 10 through 18 as follows:

- Section 10.0 - AutoMax Processor Overview - Describes the AutoMax Processor module and online operation of the AutoMax distributed control system.
- Section 11.0 AutoMax ON LINE Menu - Describes how to load, run, stop, monitor, and modify application tasks in the rack. Also describes Processor error logs.
- Section 12.0 - ON LINE Menu: Connect - Describes how to change the baud rate, enter/release the password, and enable and disable AUTO RUN of application tasks.
- Section 13.0 - ON LINE Menu: Info/Log - Describes how to display information about the system software on any Processor or UDC module in the rack and how to view the status and error log for tasks.
- Section 14.0 - ON LINE Menu: Transfer - Describes procedures for changing the default path or viewing the contents of the default system. Also contains procedures for loading tasks onto the rack and saving application tasks from the rack.
- Section 15.0 - ON LINE Menu: Running Tasks - Describes how to run tasks.
- Section 16.0 - ON LINE Menu: Stopping Tasks - Describes how to stop tasks.
- Section 17.0 - ON LINE Menu: Deleting Tasks - Describes how to delete tasks from the rack.
- Section 18.0 - ON LINE Menu: Monitoring and Editing Tasks - Describes how to monitor and change status of variables and I/O points, display and modify Ladder Logic sequences in real time, and force and un-force variables.
- Section 19.0 - ON LINE Menu: Error Clear - Describes how to clear the Processor or UDC error log or errors displayed on the Processor LEDs.
- Section 20.0 - Software Troubleshooting - Describes typical errors that can occur and procedures for correcting these errors.

6.0 CONFIGURING RACKS

The Rack Configurator is used to configure racks. With the Rack Configurator, you can create a graphical representation of your hardware installation, both the local (master) rack, and the remote (slave) rack or Remote I/O Head. The actual physical configuration of all modules in each rack and remote I/O system will be shown on the screen, with model numbers and actual names displayed where possible.

You can add modules to a rack, display or modify information about existing modules, move modules to other slots in a rack, remove modules from a rack, and enter DPS drive parameters in the Rack Configuration. Always refer to the instruction manual for the hardware you are configuring for specific configuration information. The Rack Configurator will prompt for details about the module, e.g., drop number and drop depth for Network modules, if applicable.

The information entered through the Rack Configurator is added to the database only and does not affect either application tasks or the modules themselves. For example, the switches on the faceplate of Interface modules must still be set physically on those modules even though the Rack Configurator prompts for this information to be added to the database and shows the switch setting on the screen after it is added.

You can access the Rack Configurator by selecting Configure Rack from the Rack menu in the System Configurator. An empty AutoMax rack will be displayed on the initial screen when you first start.

The Variable Configurator, which is used to map common memory and I/O to variable names, is accessible from the Rack Configurator menu. Note that the Variable Configurator is described in section 7.

See figure 6.1 for the Rack Configurator menus.

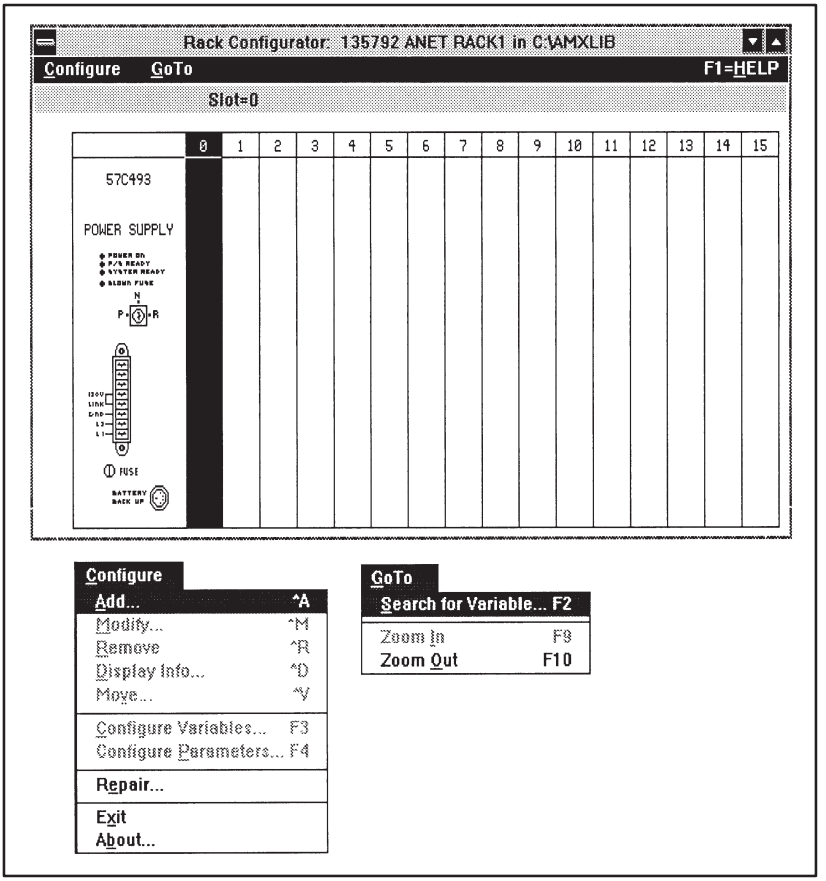


Figure 6.1 - Rack Configurator Menus

The remainder of this section describes the Configure and GoTo menu options as follows:

6.1 Adding a Module to a Rack (Configure Menu)

6.2 Adding a Remote I/O Network (Configure Menu)

6.3 Adding a Foreign Module or Other Unsupported Module (Configure Menu)

6.4 Configuring Drive Parameters

6.5 Modifying Module Information (Configure Menu)

6.6 Removing a Module from a Rack (Configure Menu)

6.7 Moving a Module to Another Slot in the Rack (Configure Menu)

6.8 Displaying Module Information (Configure Menu)

6.9 Repairing a Rack Configuration (Configure Menu)

6.10 GoTo Menu

6.11 Accessing the Variable Configurator (Configure Menu)

6.1 Adding a Module to a Rack

The Add command on the Configure menu allows you to add modules to a rack. See figures 6.2 and 6.3 for the screens displayed when you are adding a module. Follow the procedure below to add a module to the rack.

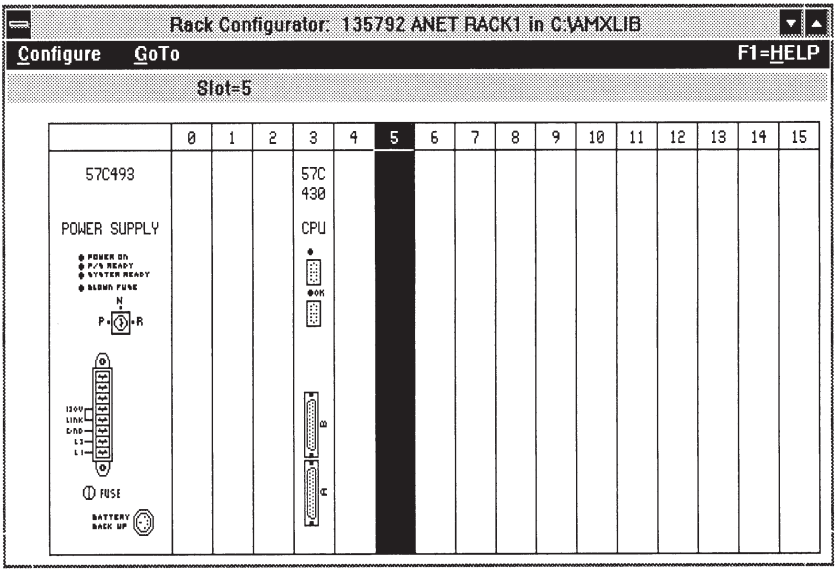


Figure 6.2 - Adding a Module, Part 1

Step 1. Select an empty slot in the rack.

Step 2. Select Add from the Configure menu. A list box with the available modules will be displayed. The first column is a two or three letter code for the module. The second column is the model number of the module. The third column displays the name of the module.

Step 3. Select the module you want to add to the rack. The modules listed below will require you to enter additional information.

M/N 57C430A, M/N 57C431, M/N 57C435
AutoMax Processor Module
When you are adding a Processor module, you will be prompted for the tick rate. The “programmable tick rate” allows you to change the definition of the tick. Changing the tick, changes the time base for tasks. This change allows you to run a task based on a unit of your choice. The programmable tick rate can be set in increments of 0.5 msec. between 0.5 msec and 10.0 msec. For compatibility with earlier Programming Executives, the default tick rate is 5.5 msec. The tick rate is defined separately for each Processor in a rack. The tick rate is transferred when the configuration object code is transferred to the Processor. The tick rate is set on the Processor as soon as the configuration is loaded. See the

AutoMax Processor Module instruction manual (J-3650) for additional information.

M/N 57C404 Network Module
If you are adding a Network module, you will be prompted for a "Network" name in addition to the drop number and depth parameters specified in the Network module instruction manual. The Network is a single letter code that serves as the name for the network.

M/N 57C440 Ethernet Module
If you are adding an Ethernet module, you will be prompted for a logical slot number (the actual slots where Multibus will access the module) when you add the module to a physical slot in the rack. The logical "slot" is either 2 and 3 or 4 and 5. Two slots are needed because the module has 128K of memory and each logical rack slot can access only 64K. Note that there are explicit restrictions about slot location in the Ethernet module manual.

M/N 57C413 Common Memory Module (128K)
M/N 57C423 Common Memory Module (256K)
A Common Memory module can be added to slot 0 in the rack, where it provides storage for common memory variables and bus arbitration for multiple Processor modules. If M/N 57C413 is being used, slot 1 must contain a Processor module, an Ethernet module (configured to use logical slots 2-3 or 4-5), or be empty, because 2 logical slots are required by the Common Memory module. If M/N 57C423 is being used, slots 1-3 must contain Processor modules, an Ethernet module (configured to use logical slots 4-5), or be empty because 4 logical slots are required by the M/N 57C423 Common Memory module.

Either version of the Common Memory module can be used in other even-numbered slots in the rack for memory storage, but in this case it must be added to the rack as two M/N GEN32K generic modules (see section 6.3). Note that M/N 57C423 provides 128K of memory storage, the same as M/N 57C413, when used in an even-numbered slot.

M/N 57C416 Remote I/O Interface Module
See section 6.2 for additional information required to configure the remote I/O system after you have added the Remote I/O module.

Foreign Module or Other Unsupported Module
If the module you want to add is not supported, i.e., not on the list of available modules, see section 6.3.

M/N 57C418 AB Interface Module or
M/N 57C417 AutoMate Interface Module
If you are adding an AB Interface module or AutoMate Interface module, you should enter the drop number as a decimal value. (The same decimal number must be physically set on the faceplate on the actual module.) The module will convert switch settings to octal values internally.

M/N 57C429 R-Net Network Interface Module
If you are adding an R-Net Interface module, you must enter the node number as a hexadecimal value. (The same hexadecimal number must be physically set on the faceplate on the actual module.)

M/N 57C424 MaxPak III Serial Interface Module
If you are adding a MaxPak III Serial Interface Module, you must enter the drop number. (The same drop number must be physically set on the faceplate on the actual module.)

B/M 57552 Universal Drive Controller Module
The Universal Drive Controller (UDC) module (B/M 57552 and B/M 57652) is used to provide drive control in Distributed Power Systems (DPS). Up to 10 UDC modules can be added to an AutoMax rack. UDC modules can be mixed with other Reliance drive control modules in the same rack. A UDC module can be added to any slot in the rack except slot 0 and those slots already designated as logical slots for an Ethernet module or a Common Memory module. When you add a UDC module to the rack, you must also select the type of drive you want to connect to ports A and B on the module. See the Configuration and Programming instruction manual for your specific drive and regulator type for instructions on how to configure the UDC module and its associated drive components. You can begin entering drive parameters by selecting the Configure Parameters command. See section 6.4 for more information. Note that you will only be able to add B/M 57652 to the rack beginning with V3.5 of the Executive software. B/M 57652 can be used in the rack configuration regardless of whether you have B/M 57552 or B/M 57652 in the rack itself.

Step 4. Select one of the following: Add Next if you want to add this module and then add another module in the next available slot; Skip Slot if you want to add this module and then leave the next slot empty; OK to add the module and return to the Rack Configurator window; Cancel to return to the Rack Configurator window without adding a module.

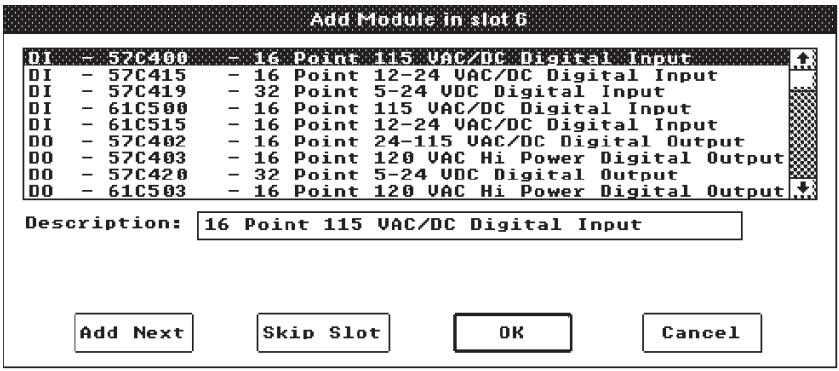


Figure 6.3 - Adding a Module, Part 2

6.2 Adding a Remote I/O Network

The following procedure enables you to add a remote I/O network for a Remote I/O Interface module (M/N 57C416). Remote racks (AutoMax or Shark), Remote I/O Heads, Local I/O Heads, Analog Rails, and I/O Rail modules are all configured using this procedure.

Step 1. Select the Remote I/O Interface module to which you want to add a remote I/O network.

Step 2. Select Zoom In from the GoTo menu. The remote I/O network diagram will be displayed.

Step 3. Select the drop on which to add a remote rack or Head.

Step 4. Select Add from the Configure menu. A dialog box with the available choices will be displayed.

Step 5. Select a remote rack (AutoMax or Shark) or Remote I/O Head. If you select a Shark rack, you must also select a power supply (115/230VAC or 24VDC) for the rack.

Step 6. Select one of the following:

Add Next if you want to add the rack or Head and then add another rack or Head on the next available drop. If you added a Remote Head in step 5, you will go to the first available port where you can configure the Head (see 6.2.2).

Skip slot if you want to leave this slot empty and move to the next available slot.

OK to add the rack or Head.

Cancel to return to the remote I/O network diagram without adding the rack or Head.

Step 7. When you are done, select Zoom Out from the GoTo menu to return to the master rack.

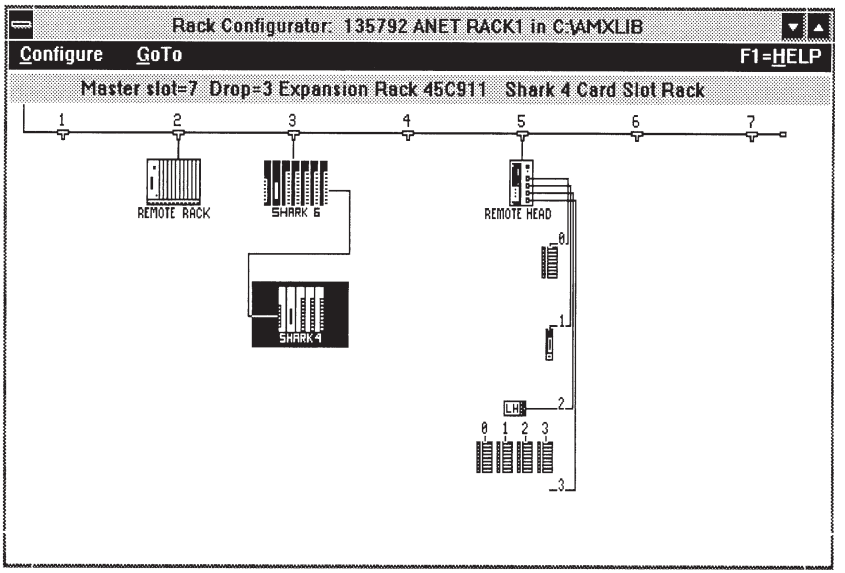


Figure 6.4 - Remote I/O Network Diagram

6.2.1 Configuring a Remote AutoMax Rack

The following procedure enables you to add modules to a remote AutoMax rack.

Step 1. Select the remote AutoMax rack you want to configure.

Step 2. Select Zoom In from the GoTo menu. A rack will be displayed containing a Remote I/O module in slot 0.

Step 3. Select an empty slot in the rack.

Step 4. Select Add from the Configure menu. A list box with the available modules will be displayed.

Step 5. Select the module you want to add to the rack.

Step 6. Select one of the following:

Add Next if you want to add the module and then add another module in the next available slot.

Skip Slot if you want to leave this slot empty and move to the next available slot.

OK to add the module and return to the Remote Rack diagram.

Cancel to return to the Remote Rack diagram without adding a module.

Step 7. When you are done, select Zoom Out from the GoTo menu to return to the remote I/O network diagram.

6.2.2 Configuring Remote Heads

The following procedure enables you to add Local I/O Heads, digital I/O rails, or Drive Interface modules to ports on a Remote Head.

Step 1. Select a port on the Remote Head you want to configure.

Step 2. Select Add from the Configure menu. A list box with the available choices will be displayed.

Step 3. Select a Local I/O Head, a digital I/O rail, an analog rail, or an Drive Interface module.

Step 4. Select one of the following: Add Next if you want to add the module and then configure the next port; Skip if you want to leave the port empty and move to the next available port; OK to add the selection; or Cancel to return without adding the selection.

Step 5. If you added a Local Head, you can then add a digital I/O rail or an interface module to the local ports.

Step 6. If you added a digital I/O rail, you can zoom in to configure the modules in the rail.

Step 7. When you are done, select Zoom Out from the GoTo menu to return to the Remote I/O Network diagram.

6.2.3 Configuring a Remote Shark Rack

Use the procedure that follows to add an expansion rack and modules to a remote Shark rack. Only one expansion rack can be added to a main Shark rack, regardless of the size of the main rack. The main rack will contain a Shark Remote I/O Interface module in slot 0 (adjacent to the Power Supply). Therefore, a main rack with "n" slots will have "n-1" slots available for I/O modules. All of the slots in the expansion rack can be used for I/O modules. However, only the first 10 slots of the main and expansion racks combined can contain I/O modules.

Step 1. Select the remote Shark rack you want to configure.

Step 2. If you want to add an expansion rack, select the end of the cable that is connected to the Shark rack. An empty square will be highlighted.

Step 3. Select Add from the Configure menu. A list box with the available Shark racks will be displayed.

Step 4. Select an expansion rack from the list, and then select a power supply for the expansion rack.

Step 5. Zoom into the main Shark rack. A rack will be displayed containing a Shark Remote I/O Interface module in slot 0.

Step 6. Select an empty slot in the rack.

Step 7. Select Add from the Configure menu. A list box with the available modules will be displayed.

Step 8. Select the module you want to add to the rack.

Step 9. Select one of the following:

Add Next if you want the module and then add another module in the next available slot.

Skip if you want to leave the slot empty and move to the next available slot.

OK to add the module and return to the Remote Rack diagram.

Cancel to return to the Remote Rack diagram without adding a module.

Step 10. Configure the expansion rack in the same manner as the main Shark rack. Note that the slots in the expansion rack are numbered beginning at one greater than the highest slot number in the main rack. Slot 10 is the last slot that may contain an I/O module.

Step 11. When you are done, select Zoom Out from the GoTo menu to return to the remote I/O network diagram.

6.3 Adding a Foreign Module or Other Unsupported Module

Any foreign module, i.e., non-AutoMax or DCS 5000, must meet strict requirements before it can be used in an AutoMax rack and configured for the rack. See Appendix H for these requirements before attempting to add a foreign module to the rack.

Foreign modules and AutoMax modules that are not supported in this version of the Programming Executive software can be added to the rack as “generic” I/O modules. The generic module should also be used if a currently existing module is enhanced and the existing form does not allow you to configure the view or changed registers. Three generic modules are available, one with 32 registers (GEN32), one with 32,768 registers (GEN32K), and one with 8000H registers in hex (GEN32KH). GEN32K and GEN32KH are available only in the local rack. GEN32 is available in a local rack and in a remote rack.

6.4 Configuring Drive Parameters

The Configure Parameters command on the Configure menu is used to configure drive parameters. This command can be selected only after you have added a Universal Drive Controller (UDC) module to the rack. The UDC Parameter Entry screens can be accessed in two ways. From the main Rack Configurator screen, select the UDC module and then select Configure Parameters from the Configure menu. From the Power Module Interface display screen (accessed by zooming into the UDC module), select Configure Parameters from the Configure menu. Refer to the DC Drive Configuration and Programming instruction manual (S-3006) for detailed descriptions of how to configure the drive parameters.

6.5 Modifying Module Information

The following procedure enables you to modify information for the module in the selected slot. See the section below if you want to modify UDC module information. The Modify procedure does not allow you to remove the module from the rack; it only allows you to modify information in the database about the module in the selected slot.

Note carefully that any variables already defined for the module in the Variable Configurator described in section 7 will remain unchanged.

Step 1. Select the slot containing the module for which you want to modify information.

Step 2. Select Modify from the Configure menu. A Card Info dialog box will display the information for the requested module.

Step 3. Modify the desired fields. Any field that is not dimmed can be changed.

Step 4. Select OK to accept changes or Cancel to return to the Rack Configurator window without executing the changes.

Modifying UDC Module Information

Note that UDC modules cannot be changed to another type of module. If the module is added by mistake, it must be removed before another module can be added. If the drive type of the UDC is changed, the PMI hardware and the parameter information will be reset to the default state of the new drive type.

6.6 Removing a Module from a Rack

The following procedure enables you to remove a module from the rack. Note carefully that any common memory variables or I/O configured for the module will be deleted from the corresponding rack database file. For UDC modules, all attached hardware and parameter descriptions will also be deleted.

If a Remote I/O module is removed from a rack, all of the remote racks, heads, modules, and I/O in the remote racks will be deleted as well.

If a Universal Drive Controller module is removed from a rack, the Power Module Interfaces which are attached to it and all its associated configuration information (hardware and variables) will also be deleted.

If you remove an AutoMax Processor or the Common Memory module (in slot 0) from a rack, the common memory variables configured for the rack will not be deleted until the last remaining Processor or Common Memory module is removed. When the last Processor or Common Memory module is removed, all common memory variables in the rack will also be deleted.

Step 1. Select the module you want to remove from the rack.

Step 2. Select Remove from the Configure menu. A dialog box will be displayed for you to confirm the deletion. The module will be removed after confirmation, and its slot in the rack will be empty.

6.7 Moving a Module to Another Slot in the Rack

The following procedure enables you to move the selected module and all hardware attached to it to another slot in the rack. For remote Shark racks, you can move modules from the main Shark rack to the expansion rack, or vice versa. If the new slot contains a module, the modules will swap locations. All of the variables associated with the module(s) are moved with the module(s).

Step 1. Select the module you want to move.

Step 2. Select Move from the Configure menu. A dialog box will display the module's present slot and the cursor will appear in the New slot field.

Step 3. Enter the new slot for the module. If the slot you enter is occupied by another module, the modules will swap slots.

Step 4. You have the option of backing up the database(s) for the module(s) being moved before they are moved. It is recommended that you do so.

The backup procedure creates a copy of the rack, module, and variable databases in a subdirectory called AMXWORK on the drive designated for temporary working files during the AutoMax Setup procedure. Note that the AMXWORK subdirectory name is reserved by the Programming Executive. If you select to have the databases backed up and an error occurs during the move, you can use the Repair command to correct the rack configuration. See 6.9 for the Repair procedure.

Step 5. Select OK to begin the move or Cancel to return to the Rack Configurator window without executing the changes.

6.8 Displaying Module Information

The following procedure enables you to display the information for the module in the selected slot, but no changes can be made.

Step 1. Select the module for which you want information displayed.

Step 2. Select Display Info from the Configure menu. A dialog box will display information in the database about the selected module.

Step 3. Select OK to return to the Rack Configurator window.

6.9 Repairing a Rack Configuration

Errors can occur if variables are found in an empty slot or if a module or variables are assigned to non-existent slots. These errors will not be evident until you try to generate the configuration file for downloading to the Processor(s) in the rack or you perform a Verify operation on an application task.

Errors like these can result when a rack is created by the Import procedure, or if the database files are modified outside of the AutoMax Executive software. Any variables (and the module, if present) or local variables with unresolved network-wide names can be removed by using the following procedure.

The Repair option should be used only if the Programming Executive software displays an error message notifying you of an error. Otherwise, this option works in exactly the same way as Remove, which deletes a module and the associated variables.

When you use the Repair option, you will need to enter the slot number referred to in the error message on the screen. If the message refers to slot 99, there has been an error involving common memory variables, which are stored in slot “99” for the purposes of the database. When these variables are configured in the Variable Configurator, they are actually mapped into slots containing Common Memory modules or Processor modules.

Use the following procedure to repair a configuration.

Step 1. Select Repair from the Configure menu. The Repair dialog box will be displayed.

Step 2. Select one of the following options:

Delete modules and variables - Deletes the module and all assigned variables for the designated slot.

Delete locals with unresolved network-wide names - deletes Network module variables (the type listed in the Local column in the Variable Configurator for Network modules) that have network-wide names that cannot be found in the network database.

Copy database backups from work directory - copies databases that were stored in the AMXWORK directory during an unsuccessful move back to the currently selected rack's directory.

Step 3. Enter the slot number to which the variable(s) are attached as noted in the error message.

Step 4. Select OK to repair the slot or Cancel to return to the Rack Configurator window without executing the changes.

6.10 GoTo Menu

The GoTo Menu is used to search by name for variables mapped to modules in the rack. It is also used to zoom in and out of the three levels (local rack<-->remote rack or head<-->digital rail) of configuration forms in the Rack Configurator and to access the Variable Configurator from the Rack Configurator.

6.10.1 Locating a Variable by Name

Select Search for Variable from the GoTo menu in the Rack Configurator to locate a variable by name. Follow the procedure below to search for a variable.

Step 1. Enter the variable name (no subscripts).

Step 2. Select the option button indicating whether the variable is a local variable or a network-wide variable.

Step 3. Select OK to begin the search. A dialog box will be displayed indicating if a match was found, the type and location of the variable found, and the slot number.

Step 4. Select Yes if you want to search for another match. A dialog box will tell you when there are no additional matches. Selecting OK will return you to the Search for Variable dialog box. You can repeat the search or change the variable name or search mode. Select No to display the form containing the variable name requested. The variable name will be selected.

The search procedure will take you into the Variable Configurator. You can return to the Rack Configurator by zooming out (see section 7.6 for additional information). However, note that upon returning to the Rack Configurator, the module that is selected will be the same one selected before the Search was performed, rather than the module containing the variable that was searched for.

6.10.2 Zooming In and Zooming Out

The Zoom In command will move you to the next lower level (greater detail). If there is another hardware level below the selected module (such as for the Remote I/O module (M/N 57C416), which "contains" the remote I/O network, or the DIgital Rail (M/N 45C1), which contains 2-bit I/O modules), zoom in will always take you to that view. At the

lowest hardware level (all other modules), zoom in will access the Variable Configurator described in section 7. To access the Variable Configurator for either the Remote I/O or the Digital Rail modules, use the Configure Variables command, which always accesses the Variable Configurator, regardless of the module selected. The Zoom Out command reverses the process, taking you back to the next higher level. When using a mouse, double-clicking the left button on the module performs a Zoom In command on that module, and double-clicking the right button performs a Zoom Out command. See figure 6.5.

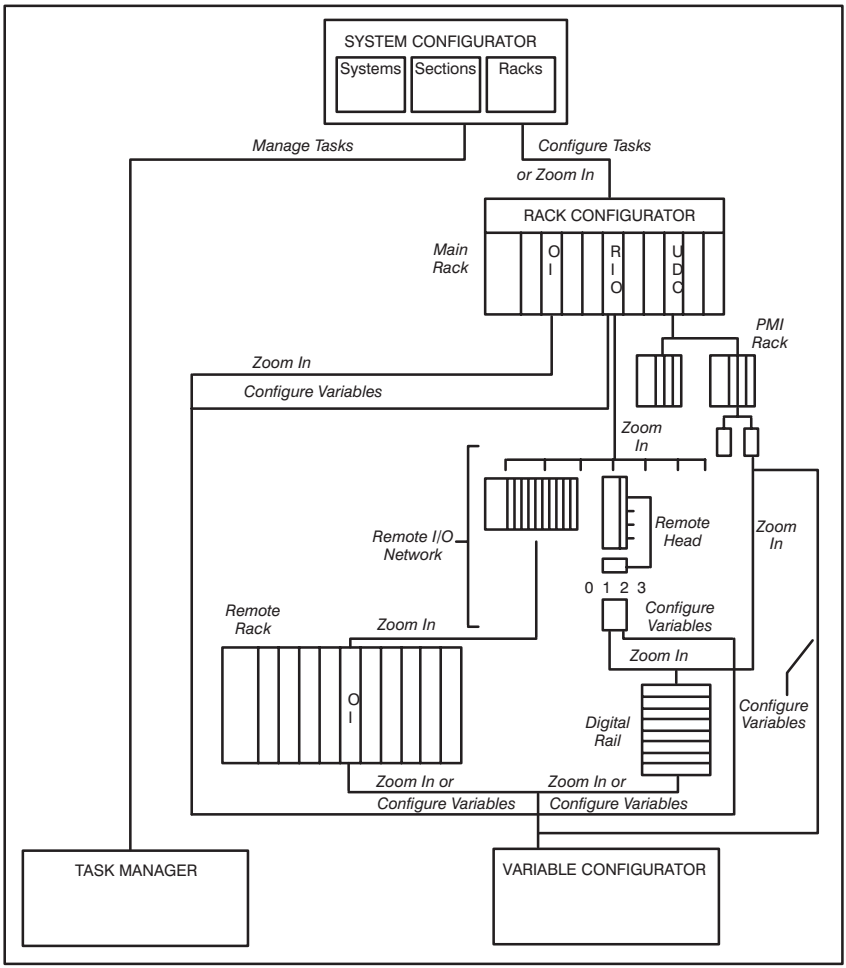


Figure 6.5 - Zoom In

6.10.3 Selecting a Drive (DPS systems only)

If you are configuring a Distributed Power drive system (DPS), the GoTo menu will provide two additional options (Drive A and Drive B) after you have zoomed into a UDC module. The Drive A and Drive B commands can be used to select PMI A or PMI B so that you can you can add analog or digital rails, thumbwheel switches, or LED modules. See the DPS Configuration and Programming instruction manual (S-3006, S-3015, or S- 3016) for detailed information.

6.11 Accessing the Variable Configurator

After you have configured a rack, select Configure Variables from the Configure menu to access the Variable Configurator. Use the Variable Configurator to assign variable names for I/O on a form designed specifically for each module. Section 7 describes the Variable Configurator.

7.0 CONFIGURING VARIABLES

The Variable Configurator is used to configure I/O points and common memory variables in the rack. The Variable Configurator allows you to “map” names to bits and registers on I/O modules and interface modules in the rack, as well as to common memory variables on Processor and Common Memory modules in the rack.

Variables are configured using “forms” (screens) that are specific to each module. Mapping variables to physical locations on individual modules and to common memory allows the programmer to create application programs referencing variable names instead of actual physical locations. All variable configuration data is downloaded to the Processors in the rack before application tasks are put into run.

After selecting a module to configure and choosing Configure Variables from the Configure menu in the Rack Configurator (or double-clicking the left mouse button on the desired module), the Programming Executive will display the configuration form for that module. Names for registers (and bits where applicable), as well as descriptions, are entered in the fields displayed at the bottom of the screen.

The variable configurator allows you to define bits (booleans), integers (16 bits), double integers (32 bits), and real variables (32 bits). The bits within a register that is already defined may be defined individually. **Note that in this case neither the bits nor the register can be forced while online.** With the exception of registers on generic modules, no single integer variable can also be defined as part of a double integer variable. I/O points cannot be defined as real variables. Refer to the programming language instruction manuals for more information on variable types.

The remainder of this section is organized as follows:

7.1 Variables That Must Be Configured

Variables that must be accessible to more than one task in the rack or that are connected to a field I/O signal must be defined in the Variable Configurator. These variables are called “common” variables. They can be accessed by all application tasks in the rack that declare them COMMON. Both I/O variables and common memory variables, i.e., those not connected to field signals, are common variables in this sense and both must be defined in the Variable Configurator. Local variables, those variables that do not need to be accessible to more than one task in the rack, and that are not connected to a field signal, are not configured in the Variable

Configurator. Local variables are defined only in the tasks that use them.

Note**:** *Although the following modules themselves can be added to the rack, registers on these modules cannot be configured in the rack:*

M/N 57C405	DCS Drive Analog I/O module
M/N 57C408	DCS Drive Power Module Interface module
M/N 57C440	Ethernet Network Interface module
M/N 61C605	8-Channel Thermocouple Input module
M/N 61C613	16-Channel Analog Input module

See 7.1.1 for more information on configuring I/O variables and 7.1.2 for configuring common memory variables.

7.1.1 Configuring I/O Variables

When you configure I/O or interface modules (except the Network Interface) module, the register/point locations are displayed in the form in numerical order. Assigning the name P_BUTTON@ to bit 2 of register 0 on a digital input module form, for example, attaches the name P_BUTTON@ to the value stored in bit 2 of register 0, which is physically connected to external hardware by way of a connector on the module faceplate.

For I/O modules with duplicate registers that indicate the same status but differ only in that the bits on one are chiefly R (read-only) and bits on the other are chiefly R/W (read/write) in the instruction manual, you may configure only the bits that are read only in the read only register and the bits that are read/write in the read/write register. An example is the 5V-24V DC Input module (M/N 57C419). For this type of register arrangement, R/W refers to the ability of the operating system, not the application task, to write to that register.

Most I/O and interface modules are configured using the basic form described in 7.2.1. Network Interface modules are configured using a special form described in 7.2.3.

7.1.2 Configuring Common Memory Variables

The common memory variables in the rack are mapped to the Common Memory module in slot 0 and Processor modules. These variables are strictly memory variables and are not connected to external hardware. There is no relationship between the way the variables are displayed in the form on the screen and the way in which they are stored. Common memory variables can be made non-volatile, i.e., their last value is preserved in the event of a power outage, if the non-volatile option is selected when they are being configured. Note that the method by which common memory variables are stored and the form used to configure them are unique. See 7.2.2 for more information.

7.2 Forms Used for Configuring Variables

There are three types of forms used to configure variables, a basic form used for I/O and interface modules, a form for common memory variables, and a form used for Network Interface modules. These forms are described in sections 7.2.1 - 7.2.3 below.

7.2.1 Basic I/O and Interface Module Configuration Form

The basic form is used for I/O modules, interface modules (except the Network Interface; see 7.2.3), and UDC modules. The basic form is essentially a list of the registers/bits that can be configured for that module. Some forms might include descriptions for dedicated bits or registers, e.g., “interrupt enable.”

The basic form is also used to configure Universal Drive Controller (UDC) modules (M/N 57552). There are eight views available for the Universal Drive Controller module form. Refer to the DC Drive Configuration and Programming instruction manual (S-3006) for a detailed description of the configurable variables on the UDC module.

7.2.2 Common Memory Variable Configuration Form

A second type of form in the Variable Configurator allows you to configure common memory variables. Each rack has one single form for configuring the common memory variables in that rack. The form is organized by item number, not by register/bit location because storage of common memory variables is determined by the Processor module(s) and Common Memory module, if any, in the rack. This single form is accessible from all Processor modules and from the Common Memory module. For example, if you configure item 64 on the Processor module in slot 2 as VAR_1%, the form for any other Processor or the Common Memory module in will display VAR_1% as item 64. If the non-volatile checkbox is checked, the variable will retain its value in the event of a power loss or power cycle.

Because common memory variables are not assigned to a module in any one slot, they are actually stored in the database for the rack in slot 99, regardless of the module on which they were originally configured. Slot 99 is not an actual physical slot in the rack and cannot therefore be viewed by the user. It may, however, be referenced in an error message on the screen, or after a version 2.0 rack is imported.

The following common memory variables are pre-defined for every rack. However, they do not appear on the form for common memory variables. You must enter these variable names on the form if you want to use these variables in application tasks.

AUTORUNSTATUS@ -	True when AUTO RUN is enabled for the rack; false if AUTO RUN is not enabled
FORCINGSTATUS@ -	True when a variable is forced in the rack; false when no variables are forced in the rack

BATTERYSTATUS0@ - True when the on-board battery of the Processor module or Common Memory module in slot 0 is OK

BATTERYSTATUS1@ - " " " " " " " " " " " " 1 " " "
BATTERYSTATUS2@ - " " " " " " " " " " " " 2 " " "
BATTERYSTATUS3@ - " " " " " " " " " " " " 3 " " "
BATTERYSTATUS4@ - " " " " " " " " " " " " 4 " " "

7.2.3 Network Interface Module Configuration Form

The third type of form in the Variable Configurator is used to configure Network Interface modules (M/N 57C404). This form is more complex than the other two forms because of the nature of the Network module. There are four "views" available for the Network module form (see 7.3 for a description of views).

Two views, Drop Status and Setup, are identical to the form used for basic I/O (described in 7.2.1.). These views allow you to configure, respectively, the drop status and setup registers for the selected Network module.

The Drop Areas 1-55 view and the Broadcast Data view allow you to assign names and descriptions to all data registers in all drop areas (1-55) on the network. The drop area number being configured will be displayed on the screen. The number of drop areas controlled by the currently-selected Network module is determined by the module's drop depth, which was set when the module was added to the rack. Registers that can be read/written in these drop areas (0-31) show "R/W" before the register number. Registers 32-63 in drop areas controlled by this Network module, as well all other registers in all other drop areas on the network, are shown preceded by an "R" because they are read-only for tasks in the currently selected rack. Note that if the drop number of the Network module being configured is 0, i.e., the drop is the master, then registers 0-31 are displayed as read only and register 32-63 are displayed read-write for all drops 1-55. See figure 7.1 for a sample screen in the Drop Areas 1-55 view.

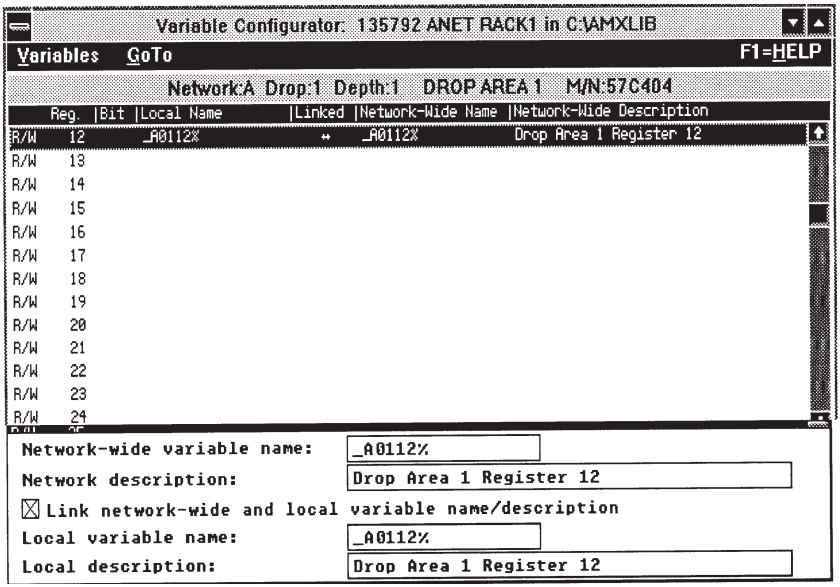


Figure 7.1 - Configuring Drop Areas 1-55 in Network Interface Modules

You can assign two names (and two descriptions) to each register/bit in the Drop Areas 1-55 view and the Broadcast Data view: a “network-wide” name and a “local” name.

The network-wide name for a register/bit is effectively a label that identifies or “reserves” the register/bit location across the entire network and can be assigned for all the registers/bits from any Network module on the network. Typically, network-wide names for all registers/bits in the network are assigned first to provide a template for configuring the individual drops.

When a network-wide name is assigned to a register/bit, it is placed in a database ($NET.DBF, one per system). The network-wide name will be displayed in the “network-wide name” field for that register/bit location on every Network module on the network. This keeps the register/bit location from appearing unused in other Network modules and from being inadvertently assigned another variable name, e.g., “speed_feedback%” is assigned to register seven in drop area 15 in one rack and (by mistake) to register eight in drop area 15 in another rack.

Although assigning it a network-wide name defines the register/bit across the network, it does not actually add the network-wide name to the configuration database of any rack on the network. Registers/bits that are not in the configuration database of a specific rack, i.e., not configured, cannot be accessed by application tasks in that specific rack.

In order for application tasks in the rack to be able to access the variable in this rack, the register/bit must have a local name in the rack. The local name for a register/bit will be displayed (in the “local name” field) only when configuring the Network module that contains the actual drop area where the register/bit is located. The local name will not be displayed in this register/bit location when the Network drop area is viewed through any other Network module on the network. Note that the meaning of “local” in terms of the Network module configuration form is different from “local” in the sense of variable access in a single application task as described in section 7.1.

To add a network-wide name to a rack configuration so that application tasks within the rack can access it, the network-wide name must be linked to a local name. This is done by checking the “Link network-wide and local variable names/descriptions” checkbox when the desired network-wide name is selected. Note that whether the local or network-wide names are linked or not, only the network-wide description will be displayed on the form. The local description will be displayed in the default fields at the bottom of the screen when the register is selected.

If you link the network-wide and local variable names when the local name field is empty, the network-wide name will be copied to the local name field. Conversely, if the local name is specified and the network-wide field is empty, checking the box will copy the local name to the network-wide field. Network-wide and local names doe not have to be the same in order to be linked. When the names are linked, the “Linked” column on the screen will display a double-headed arrow (<– –>). If you delete a network-wide variable name, any local variables linked to that name will also be deleted for all racks in the system.

Because of the way in which linked network-wide names are stored (in the network database as well as in individual rack configurations), there is a great deal of flexibility in manipulating the configuration of the network. You can edit or copy linked network-wide names from any Network module on that network. Linking the local name to the network-wide name is also helpful in case you need to re-organize your use of network registers because all local names linked to a network-wide name are copied when the network-wide name is copied. Further, if you edit the network-wide name for a register, the new name will replace the old network-wide name for all racks on the network. It is important, however, to understand the results of editing a linked network-wide name.

In order to successfully edit, i.e., change the name of a linked network-wide variable, the rack configuration databases for all the racks on the network must be present and accessible so that the AutoMax software can change the name in each at the same time. The reason for this is that when the configuration object code for a rack is generated, the actual register/bit address of a linked local name is determined using the network-wide name, which is also stored in the rack’s database, as a key into the network database. If a register/bit has a local name which is NOT linked to a network-wide name, the register/bit address is found in the rack database, not the network database. If the change cannot be made in all of the racks’ databases at this time, the old network-wide name (in the unchanged rack’s database) will no longer have a corresponding entry in the network database. This condition will be flagged as an “unresolved network-wide name” when the configuration for this rack is generated.

Because local names are displayed only in the rack in which they have been assigned, if you mix un-linked local names with linked local names, there is a danger that a network-wide name, along with its linked local name, could inadvertently be copied on top of an un-linked local name. For this reason, it is recommended that un-linked locals and linked locals not be mixed on the same network. This problem will go undetected until either the view of this drop is called up on the Network module containing the un-linked local, or until the configuration for that rack is generated.

If you transfer racks out of the system to another location, e.g., for start-up or testing, you should not make any changes to linked network-wide variables. These changes will not be copied back to the network database when the rack is transferred back into the system. In this situation, un-linked variables can temporarily be used to implement the required changes.

7.3 Register Views and Register Pages

Variable Configurator forms are sometimes divided into “views” of registers. A view is a logical collection of variables for the particular module being configured, e.g., status registers.

When there are a large number of variables to be configured for a particular module or within a particular view, the form or the view within the form will be divided into pages of registers, all of which are accessible by scrolling.

You can move among different views of the form by selecting Change View from the GoTo menu. There may only be one view for a particular form, in which case the Change View option will be fuzzed. There may be more than one page for each view, in which case you may move forward and backward (according to register number) using Next Page and Previous Page from the GoTo menu. If there is only one page, inapplicable menu items will be fuzzed.

7.4 Entering Variable Names and Descriptions

Before configuring the registers on any module, read the instruction manual for the modules you want to configure in order to familiarize yourself with the available registers and bits on the module and any dedicated registers or bits. See figure 7.2 for the menus available in the Variable Configurator and the initial screen display.

The Variable Configurator screen display and the procedure for configuring variables are described below. The menus available from the Variable Configurator are described in sections 7.7 and 7.8.

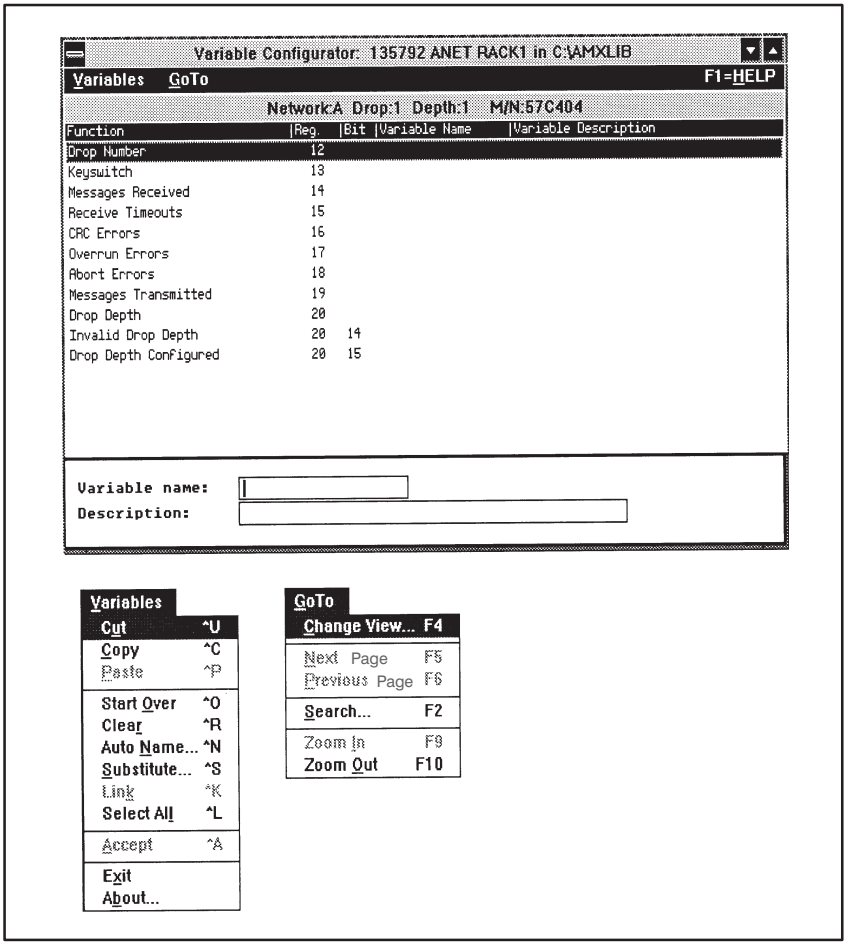


Figure 7.2 - Variable Configurator Menus

The following fields appear at the top of the form used to configure the selected module. The form used for Network modules is slightly different. Refer to section 7.2.3 for more information about the form.

Function: function of the register/bit, if any. If this field contains a name, e.g., “Interrupt Reset,” this is the dedicated function assigned to the register/bit; otherwise, the field contains a generic description plus the register/bit number

Reg: register number on I/O module or item number for common memory variables

Bit: bit number within the register or an * if bits are configured for the register. In addition, for network modules, a “+” will be displayed if the register contains global bits that are not being used in the rack.

Variable Name: the variable name assigned.

Variable Description: the description assigned to the register/bit

Item: used in place of register number for Processor and Common Memory modules

Follow the directions below to configure variables for a module.

Step 1. Select a module to configure from the rack displayed.

Step 2. Select Configure Variables from the Configure menu in the Rack Configurator (or double-click the left mouse button on the desired module). At this point, a view selection dialog box may be displayed for some modules. After selecting the view, the first page of registers/bits in the view will be displayed.

Step 3. Select a register.

Step 4. Define the register or zoom in to define the bits in the register, if applicable, by entering a name and description. Variable names are limited to 16 alphanumeric characters, and must begin with a letter or an underscore (_). Variable names must end with the proper terminating character for the variable type (% for single integer, ! for double integer, @ for boolean, $ for string, (none) for real).

Note**:** *Variables that will be used in PC/Ladder Logic application tasks are limited to 14 characters in length. Variable descriptions can be up to 40 characters in length.*

WARNING

IF YOU USE DOUBLE INTEGER VARIABLES, YOU MUST IMPLEMENT SOFTWARE HANDSHAKING BETWEEN APPLICATION TASKS REFERENCING THE VARIABLES TO ENSURE THAT BOTH THE MOST SIGNIFICANT 16 BITS AND THE LEAST SIGNIFICANT 16 BITS HAVE BEEN TRANSMITTED BEFORE THE VARIABLE IS READ BY AN APPLICATION TASK. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY OR DAMAGE TO EQUIPMENT.

Step 5. Select another location or press ENTER. The name and description entered in step 4 will be displayed for the register selected in step 3.

Step 6. When you are finished assigning variable names, you can leave the Variable Configurator by selecting Exit or Zoom Out. You will be prompted to accept the changes you have made to the database. If accepted, the variable names and descriptions will be added to the database.

Note carefully that AutoMax does not check for errors in variable type as names are assigned. If there are errors, they are usually found when you accept the changes. The database will not be updated, and an error message will be displayed. Some errors, e.g., duplicate variable names, may not be discovered until you attempt to generate the rack configuration.

7.5 Configuring I/O for a Foreign Module

Foreign modules are added to the rack as generic modules (GEN32, GEN32K or GEN32KH). To configure I/O for a generic module, select the register range and then follow the procedure described in section 7.4. Make certain that the foreign module meets the requirements in Appendix H.

7.6 Modifying Variable Names and Descriptions

Use the following procedure to modify variable names or descriptions.

Step 1. Select the register/bit that you want to change.

Step 2. Backspace over the existing text and enter the new text.

Step 3. Select another register/bit or press ENTER. The new name and/or description will be displayed.

Step 4. When you are finished changing variable names, you can select Accept from the Variables menu to add the new names/ descriptions to the database. Otherwise, when you leave the Variable Configurator, you will be prompted to save the changes.

7.7 Variables Menu

The Variables menu consists of commands that enable you to copy, delete, or modify variable names. You can also choose to have the Executive software assign variable names to selected I/O points.

7.7.1 Cutting/Copying and Pasting Variable Names and Descriptions

The names and descriptions of existing variables can be copied into the Windows Clipboard and then can be assigned to other registers or bits of the same type, i.e., integer to integer and boolean to boolean. You can cut/copy and paste variables to other registers or bits on the same module or to another module. Network register names can be pasted only into the network module form. The contents of the Clipboard will remain until replaced by new data the next time the Cut or Copy command is executed or until you leave Windows.

Variable names and descriptions can be pasted to other variables in three ways. If a single variable name and description are copied to the Clipboard, they can be pasted to either (1) another single destination or (2) to multiple destinations by selecting multiple registers. If multiple variable names and descriptions are copied to the Clipboard, they can (3) be copied only to a single destination which acts as the starting point for the paste. The contents of the Clipboard can be pasted as many times as required.

Note that when a register is copied, the bits assigned to that register are not copied. The bits for the register must be copied and pasted as a separate operation.

Note that when non-consecutive items are copied into the Windows Clipboard, they become a consecutive list. For example, if three items are selected from a list of registers (registers 1, 4, and 5 on the list) are copied into the Clipboard and then are pasted to a new destination, they will appear in consecutive order, beginning at the starting point selected for the paste operation. If you use register 1 as the starting point at the destination, the paste will result in names and descriptions being assigned to registers 1, 2 , and 3. However, if you select and copy a contiguous block of names and descriptions into the Clipboard (including any unassigned registers within the block), they will retain their relative positions when pasted to a new destination.

7.7.1.1 Copying Variable Names to Other Registers on the Same Module

Use the following procedure to copy variable names and descriptions to other registers or bits on the same module.

Step 1. Select one or more variable names and descriptions to be copied to the Clipboard. See 4.6 for instructions on making multiple selections.

Step 2. Select one of the following commands from the Variables menu:

Copy will make a copy of the highlighted variable name(s) in the Clipboard. The names and descriptions on the form will not be affected.

Cut will make a copy of the highlighted variable name(s) in the Clipboard, but it will remove the selected variable name(s) and description(s) from the form.

Step 3. Select the starting location(s) to which you want to paste variable names and descriptions.

Step 4. Select Paste from the Variables menu. The variable names and descriptions will be assigned to the selected destination.

Step 5. Use the edit boxes at the bottom of the screen to make any changes to the newly assigned names and descriptions.

7.7.1.2 Copying Variable Names to Registers on Another Module

Use the following procedure to copy variable names and descriptions from one module to another.

Step 1. Select one or more variable names and descriptions to be copied to the Clipboard. See 4.6 for instructions on making multiple selections.

Step 2. Select one of the following commands from the Variables menu:

Copy will make a copy of the highlighted variable name(s) in the Clipboard. The names and descriptions on the form will not be affected.

Cut will make a copy of the highlighted variable name(s) in the Clipboard, but it will remove the selected variable name(s) and description(s) from the form.

Step 3. Select Exit or Zoom Out to leave the Variable Configurator. You will return to the Rack Configurator.

Step 4. Select the module to which you want to paste variable names and descriptions, then select Configure I/O from the Configure menu to return to thbe Variable Configurator.

Step 5. Select the starting location(s) to which you want to paste variable names and descriptions.

Step 6. Select Paste from the Variables menu. The variable names and descriptions will be assigned to the selected destination.

Step 7. Use the edit boxes at the bottom of the screen to make any changes to the newly assigned names and descriptions.

7.7.1.3 Copying Variable Names into Task Files

After variable names and descriptions have been copied to the Clipboard, they can be pasted as text into other Windows applications (e.g., Write). Note that you cannot paste text from the Clipboard directly to non-Windows text editors such as the Norton Editor. Each variable name will be pasted as "XX01 COMMON <variable name> \! <description>". This format allows variable names and descriptions to be pasted into tasks and used for COMMON statements. Each line number will increment by one. For network variables, the description will be the network name. If the network description is empty, the local description will be used.

7.7.2 Updating the Database

You can select Accept from the Variables menu to update the database with newly added or changed variable names and descriptions. Note that error checking, specifically for duplicate variable names, is not performed until the configuration file is generated. See section 8.16.3 on generating a configuration file.

You will also be prompted to save any changes when you select Exit, Change View, or Zoom to leave the form you are working on. The database will be updated following confirmation.

7.7.3 Using the Start Over Command

Selecting Start Over will restore the contents of the edit fields at the bottom of the configuration forms to their state before changes were made. This command is available only when changes to the fields have been made, but have not been stored. (Changes are stored when you press UP, DOWN, ENTER, etc.)

7.7.4 Deleting Variable Names and Descriptions

Use the following procedure to delete variable names and descriptions.

Step 1. Select the variable name(s) and descriptions you want to delete.

Step 2. Select Clear from the Variable menu.

Note that common memory variables can be deleted from any Processor or the Common Memory module in slot 0.

7.7.5 Naming I/O Points Automatically

You can choose to have AutoMax Executive V3.x assign variable names and descriptions for one or more selected I/O points. Variable names will be assigned as follows.

All names will begin with the character string (up to 4 characters) you designate as the prefix. The remaining characters will be digits, and will follow one of the formats below.

Key to abbreviations found in the formats below:

m= master slot (for remote I/O)

s= slot

p= port (Remote Rail I/O head)

l= local port (Remote Rail I/O head)

d= drop

R =AutoMate register prefix (R-Net Network Interface)

r= register number (user register; field will consist of only the number of digits needed to define the field.)

b= bit number (for boolean variables only)

f= Allen-Bradley file letter

n= network letter

- Digital I/O, analog I/O, DCS drive modules, Resolver, Pulsetach, A-B RIO Scanner, motion control
  Local: ssr..rbb
  Remote: mmdssr..rbb
- Universal Drive Controller
  Local: ssr...rbb
  Remote: N/A
- Network
  Setup and Status registers: ssr..rbb
  Broadcast: nr..r
  Registers in Drop Areas: nddr..r
  Remote: N/A
- Common Memory
  Local: N/A
  Remote: N/A
- R-Net Network Interface
  Local: ssRr...rbb
  Remote: N/A
- Gateway (Modbus, AutoMate, Toledo Scale)
  Local: ssr...rbb
  Remote: N/A
- Gateway (Allen-Bradley)
  Local: ssfr...rbb
  Remote: N/A
- Digital Rail, TWS, LED
  Local: N/A
  Remote: mmdplbb

- Analog Rail
  Local: N/A
  Remote: mmdpr

Example: if you designate PB as the character string, a digital input module in slot 12, register 5, bit 2 would be named PB12502@.

Follow the instructions below to automatically name an I/O point: Note that the Auto Name option will write over any text in the selected register/bit location.

Step 1. Select the I/O points for which you want names assigned.

Step 2. Select Auto Name from the Variables menu. The Autoname dialog box will be displayed.

Step 3. Enter a variable prefix of 1 to 4 characters (letters and underscore only). The Executive software will assign names and descriptions for the I/O points as described above. The Description will be the same as the functional name displayed for each I/O point.

7.7.6 Substituting Variable Names

Variable names within a view can be changed by substituting characters within their names. For example, if you enter *A00 as the search string and *B99 as the replace string, variable name PBA00 would be changed to PBB99. Follow the instructions below to substitute variable names. See figure 7.3 for the screen display.

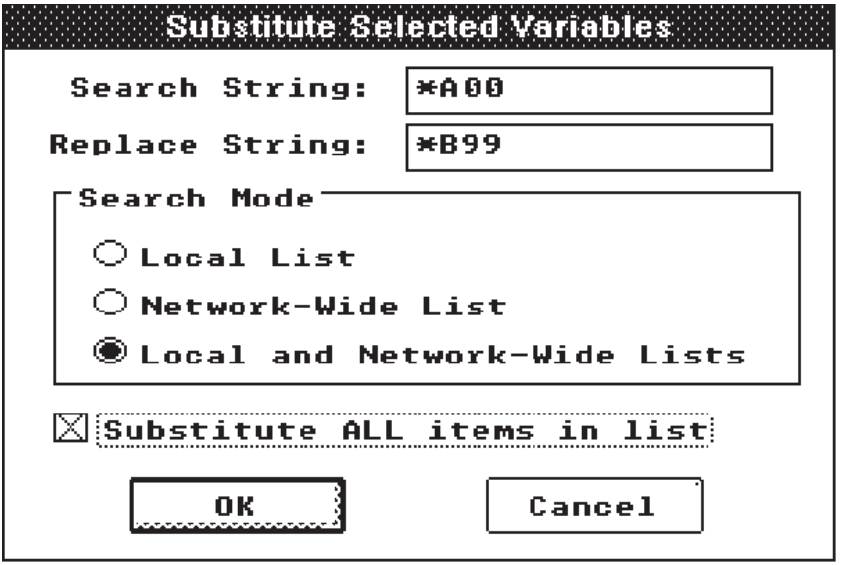


Figure 7.3 - Substitute Variable Names

Step 1. Select the variable names you want to change (unless you want all matches found; see step 5).

Step 2. Select Substitute from the Variables menu. The Substitute Selected Variables dialog box will be displayed.

Step 3. Enter the Search String and Replace String.

Search String: Can be any character string that is a part of the name of the selected variables. The wildcard character (*) can be used as the first or last character of the Search String. Do not enter the variable type character.

Replace String: Can be any character string (variable names are limited to 16 characters). If a wildcard character (*) was used in the Search String, it must also appear here in the same relative position.

Step 4. (Network module only) Select the search mode.

Local List: Substitute variable names in local list only, i.e., variables in this rack.

Network-wide List: Substitute variable names in Network-wide list only.

Local and Network-wide Lists: Substitute all variable names.

Step 5. If you want to make the substitution for all variables in the list/view that match the search string (not just the matching variables in the selected items), select Substitute ALL items in the view.

Step 6. Select OK to make the character substitution in the variable names.

Step 7. Select Accept to update the database.

7.7.7 Linking Network-Wide and Local Names

The Link command is available when the network module form with network-wide and local names is displayed. This command will link local names to network-wide names for all the selected registers. This command has the same result as selecting each item individually and checking the “Linked” option. For a given register, if there is only a network-wide name and description, it is copied to the local name and description and they are linked. If there is only a local name and description, it is copied to the network-wide name and description and they are linked. If both names exist and they are the same data type, they are linked.

Use the following procedure to link multiple network-wide and local names.

Step 1. Select the registers/bits for which you want to link network-wide and local names. See 4.6 for instructions on making multiple selections.

Step 2. Select Link from the Variables menu.

7.7.8 Selecting All Variables

Select All will select all variables in the current view. You can then execute any of the available commands (copy, clear, auto name, substitute) for all the selected variables.

7.8 GoTo Menu

Use the GoTo menu to select the type and range of variables displayed for the module during configuration. You can also search for variables by name from the GoTo menu.

7.8.1 Selecting the Variable Range Displayed

Selecting Change View from the GoTo menu displays the Change View dialog box. Choices are listed as option buttons. Use the scroll bar to scroll through the desired range of variables for the selected view.

Selecting Next Page displays the next page of variables for the selected view for the module.

Selecting Previous Page displays the previous page of variables for the selected view for the module.

Selecting Zoom In displays the boolean variables associated with the selected register.

Selecting Zoom Out at the bit level returns you to the registerlevel variables for the selected module. From the register level, zooming out will return you to the Rack Configurator.

7.8.2 Locating a Variable by Name

Select Search from the GoTo menu to locate a variable by name. This is the same variable search function as described in section 6.10.1.

8.0 CREATING AND EDITING TASKS WITH THE TASK MANAGER

The Task Manager is used to create and work with application tasks on a rack level as well as to generate the rack configuration and drive parameter files for UDC modules. You can access the Task Manager by selecting Manage Tasks from the Rack menu in the System Configurator. See figure 8.1 for the Task Manager menu structure. A list of existing tasks (if any) will be displayed, and three menus will be listed on the menu bar as follows:

Task: used to add new tasks and display or edit existing task information, verify tasks, resequence and compile tasks, transfer tasks to or from diskette, print out task listings, remove tasks, and generate cross-reference, rack configuration, and parameter object files.

Commands: used to edit DOS files, to delete files that are no longer needed, create the configuration file for a rack, and generate the drive parameter files for a rack.

On-Line**!** enables you to access the Online menu.

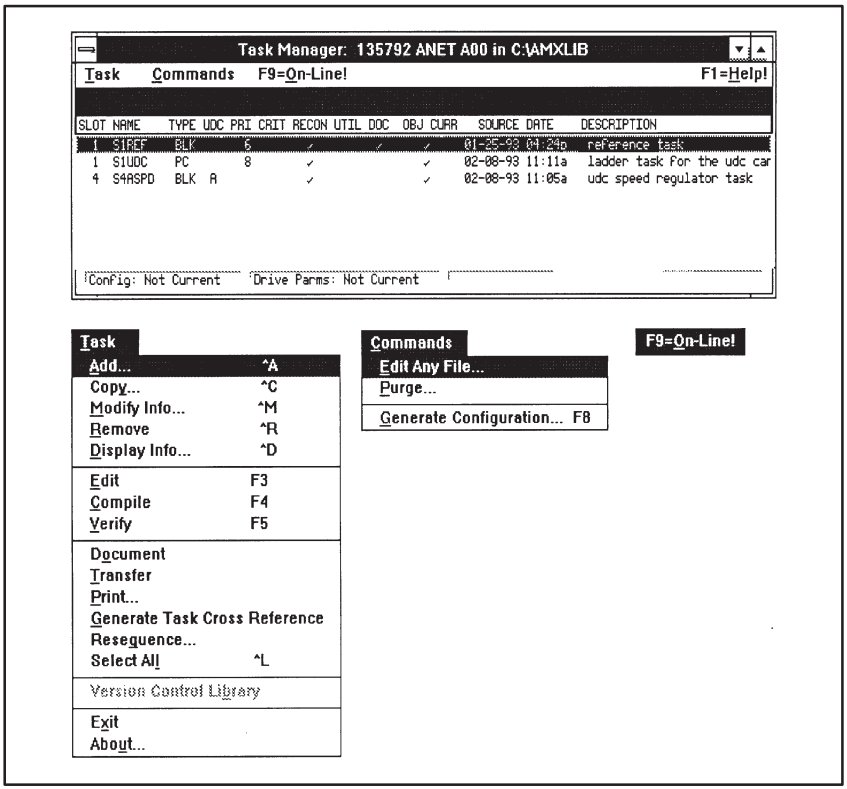


Figure 8.1 - Task Manager Menus

The fields displayed below the three menu names contain the following information for the tasks listed:

Slot - The number of the slot in the rack that contains the Processor or UDC module to which the task is assigned.

Name - Name of application task.

Type - Type of task, e.g., BASIC, using the file extension given when the task was created.

UDC - “A” or “B” to designate the drive for UDC tasks; blank otherwise.

PRI - Task priority. “4” - “11” for AutoMax tasks; blank for UDC tasks. (The task for drive A always executes first, then the task for drive B.)

Crit - Whether the critical flag has been set for the task (see 8.1.1 step 1); checkmark means “yes.”

Recon - Whether the reconstructible flag has been set for the task (see 8.1.1 step 1); checkmark means “yes.”

UTIL - Whether the task is a utility task; checkmark means “yes.”

Doc - Whether a documentation file exists for the task; checkmark means “yes.”

Obj Curr - Whether the object (compiled task) is current, i.e., whether the source date/time listed in the next column is later than the date/time the task was last compiled; a checkmark means “yes”. If the object is not current or the task has never been compiled, this column will display “NO”. The object is not current if either of the following occur after the date/time the task was last compiled: a change in the status of the “reconstructible” check box in the task information dialog box, or the task is edited.

Note that PC/Ladder Logic tasks and include (.INC) tasks do not need to be compiled and that this column will always be checked for these two types of tasks.

Source Date - Date and time the application task was last edited.

Description - Description of the task entered when the task was created.

The status line at the bottom of the Task Manager screen is used to display the status of the rack configuration file and the drive parameter file (if applicable).

The fields labeled “Config:” and “Drive Parms:” will display “Current” if the rack configuration and drive parameter files are current and the application tasks for the rack can be loaded to the AutoMax Processors and UDC modules in the rack. If either of these fields displays “Not Current”, one of the following conditions has occurred: no configuration has been generated for the rack, task information (critical flag, priority, description, etc.) has been changed after the date/time that the configuration was last generated, or the rack configuration itself has been changed (variable added or delete, task added or deleted, etc.) after the date/time that the configuration was last generated. The rack configuration, drive parameters, and application tasks should not be loaded into the AutoMax Processors and UDC modules in the rack until the configuration and parameter files are generated using the Generate Configuration command described in 8.16.3 below.

The remainder of this section is organized as follows:

8.1 Adding a Task

The first step in creating a task is to add it to the rack. Tasks cannot be created or edited without first being added to a rack using the procedure described below. This procedure can be used to add a task that will run on an AutoMax Processor or on a UDC module. See the appropriate DPS Configuration and Programming instruction manual for additional information on adding UDC tasks to the rack.

Select Add from the Task menu to display the Add New Task dialog box. Required data fields are denoted by (R) and optional data fields are denoted by (O).

Step 1. Enter information for the following fields. Some fields may contain default entries. If you do not change these fields, the default will be used.

Task name - An alphanumeric string from 1 to 8 characters long. Task names can also include underscore (_) characters. The task name must begin with an alphabetic character and it must be unique within the rack. (R)

Description - A description of the task; any character string up to 40 characters long. (O)

Engineer - Enter the name of the engineer in charge of the task. (O)

Task Type - BASIC, PC, Block, or Include. An Include task is a file to be included in a BASIC or Control Block task when it is compiled (see J-3676 for more information). The extension for each task type is shown in parentheses. Select Control Block (.BLK) for all UDC tasks. (R)

UDC Task - Available only if Block is selected as the Task Type. Select UDC Task and either A (default) or B to designate the drive.

Slot - The slot number in the rack that contains the AutoMax Processor that will run the task (0-4) or the slot number in the rack that contains the UDC module that will run the task (0-9 or 0-15, depending on the size of the rack). (R) Tasks that use Ethernet functions must be run on the left-most Processor in the rack. Refer to J-3675 for information about the Ethernet functions.

Priority - 4-11. This determines the priority with which the task will run in the AutoMax Processor. The lower the number, the higher the priority. See appendix C for more information on task priority. (R) This parameter is not available for UDC tasks.

Check boxes - Select all that apply. Note that the Critical Task and Utility Task designations are mutually exclusive.

Critical Task - specifies whether the task is critical to the system, i.e., whether it can be stopped independently (no check) or only via a STOP/ALL command from the Online menu (check). A critical task cannot be loaded into the Processor or UDC until all tasks currently in RUN status are stopped.

Reconstructible - will cause the compiler to generate an object file that supports full reconstruction when the task is compiled. If this option is not selected, tasks cannot be saved back to the personal computer from the Processor or UDC. If this option is selected, "!" comments will not be stripped from the file as it is compiled, and the object file created will be larger than it would have been if this option had not been used. The object file containing full reconstruction data (an ASCII image of the original file) is written to the application disk when the task is compiled with this option selected. This option is not available for PC tasks.

Utility Task - a task which is not specified in the rack configuration object file loaded onto the Processor module(s), but is intended to run in the rack. Utility tasks are usually used for testing purposes and not for actual applications.

Step 2. Select OK to add the task name to the list on the screen and add the task to the database, Cancel to return to the Task Manager window without adding the task.

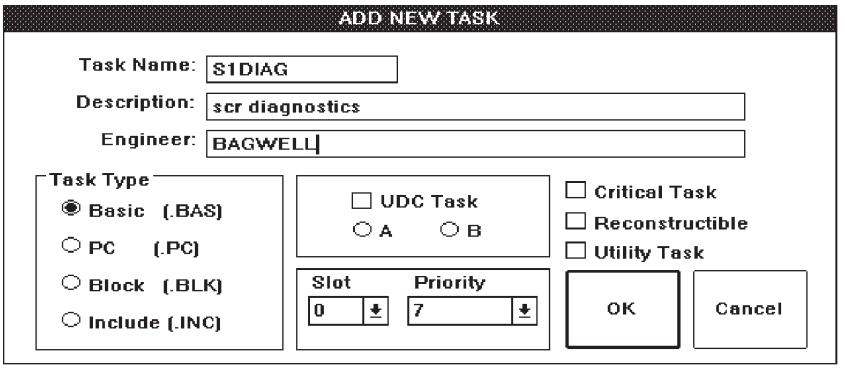


Figure 8.2 - Add New Task

8.2 Copying a Task

You can copy an AutoMax task or UDC task from any AutoMax library, system and rack into the rack you are currently working with. Select Copy from the Tasks menu to display the “Copy a Task” dialog box. See figure 8.3.

Step 1. Select the source drive and source library directory from the lists displayed.

Step 2. Select the source system, section, rack, and task names from the lists displayed.

Step 3. Select OK. The “Copy Task” dialog box will be displayed. This dialog box contains the same fields as the “Add New Task” dialog box illustrated in figure 8.2. You can modify any of the information except the task type. The task name must be unique within the rack.

Step 4. Select OK to add the task name to the list on the screen and add the task to the database, or Cancel to return to the Task Manager window without adding the task to the rack.

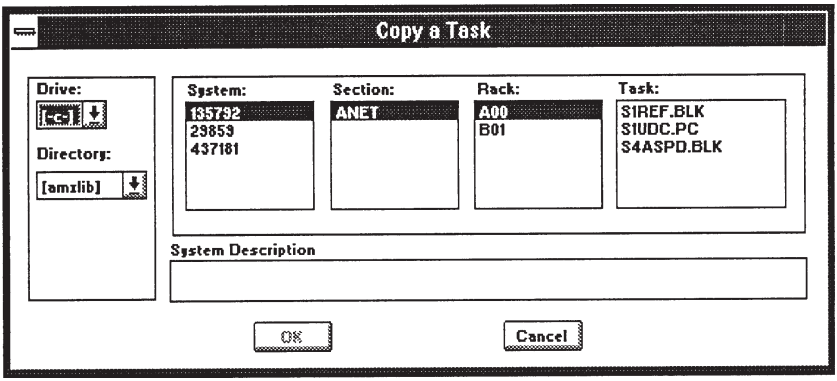


Figure 8.3 - Copy a Task Dialog Box

8.3 Modifying Task Information

Use the following procedure to modify information for a task. Any field except the task type can be changed.

Step 1. Select the task for which you want to modify information.

Step 2. Select Modify Info from the Task menu. A Task Info dialog box will display the information for the requested task.

Step 3. Modify the desired fields. You cannot modify the task type.

Step 4. Select OK to accept changes or Cancel to return to the Task Manager window without making the changes.

8.4 Removing a Task

Use the following procedure to delete the source file(s) for the selected task(s).

Step 1. Select one or more tasks you want to remove from the list.

Step 2. Select Remove from the Task menu. A dialog box will be displayed, requesting confirmation. The task(s) will be deleted from the database after confirmation.

8.5 Displaying Task Information

Use the following procedure to display the information for the selected task. You cannot make changes to any of the information displayed.

Step 1. Select the task for which you want information displayed.

Step 2. Select Display Info from the Task menu. A Task Info dialog box will display the information for the requested task.

Step 3. Select OK to return to the Task Manager window.

8.6 Editing a Task

Use the Edit command from the Task menu to edit any task. The source time and date listed for a task will be updated whenever it is edited.

Step 1. Select the task you want to edit.

Step 2. Select Edit from the Task menu to access the editor appropriate for the selected file type. The PC Editor is used for ladder logic tasks. (See section 4.22 for instructions on using the PC Editor.) The selected text editor is used for Control Block, BASIC, and Include files.

The following task skeleton files will be copied into the Executive software subdirectory as part of the AutoMax installation: SKELETON.BAS (BASIC), SKELETON.PC (PC/Ladder Logic), SKELETON.BLK (AutoMax Control Block), SKELETON.INC (Include) SKELETON.DOC (documentation files). The skeleton files will be copied into the task source file to provide a starting point for creating the task. Access your text editor from DOS to edit these files if required. Do not attempt to edit them from within AutoMax Executive V3.x.

If you have also installed a DPS software option, which contains the UDC operating systems and SKELETON.UDC file, the SKELETON.UDC file will contain the Local tunable variables that must be defined in every UDC task exactly as indicated. Note that each of the Local tunable variables in the SKELETON.UDC file is preceded by a “!” comment symbol. Simply delete the “!” symbol for each Local tunable variable required for your selected drive type. Only the “CURRENT” parameter may be edited for these Local tunable values. Other application-specific Local tunable values may be added as required. See the appropriate DPS Configuration and Programming instruction manual for more information on UDC tasks and J-3675 for more information on Local tunables.

TEMPLATE.BLK is also copied into the AutoMax subdirectory. It contains prototype Control Block statements which can be copied

into Control Block tasks to simplify their creation. This file should not be edited.

Note that when you edit a Control Block task or a UDC task, a SCAN_LOOP statement and END statement (from the template) will already be in the file when it is first opened. Both of these statements are required for Control Block or UDC tasks to be compiled. The parameters in the SCAN_LOOP statement must be filled in. BASIC tasks will contain an END statement (from the template) when the task file is first opened.

Refer to the instruction manuals for the BASIC, Control Block, and PC programming languages for more information.

8.7 Compiling a Task

Use the Compile command to compile one or more selected tasks. BASIC, UDC Control Block, and AutoMax Control Block tasks must be compiled before being downloaded to the rack. PC/Ladder Logic tasks and .INC tasks do not need to be compiled. The object code generated is written to a file with the same filename as the source, but with a file extension of .OBJ. If an object file with the same filename already exists, it will be written over.

Step 1. Select the task(s) you want to compile.

Step 2. Select Compile from the Task menu. Each of the selected files will be compiled.

AutoMax will create a log file (.LOG) for each task that is being compiled. You can view a log file by selecting Edit Any File from the Commands menu, and you can print it by using the print command from the Windows File Manager or from your text editor.

If an error is detected during compilation, no object file will be created. All errors will be listed to the screen by line number and will be written to the log file, if any. When the task has been compiled successfully, the “Obj Curr” column in the tasks display will contain a checkmark (✓).

8.8 Verifying Tasks

Use the Verify command to check that all the variables defined in the selected task(s) are used, that the configuration for the rack is valid, and that no syntax errors are present in the selected task(s).

The Verify procedure will first compile the rack configuration file, referred to as _CONF.CNF on the screen during the compilation procedure. A log file will be created to store errors, as well as compilation statistics. The log file is named _CONF.LOG. If there are any errors, an object file will not be created. Refer to Appendix D for configuration file size limits.

Note that the configuration log file errors will be displayed in the form of statements with line numbers, which are used by the operating system only. Ignore the line numbers and statement format. Use the descriptive error message immediately underneath the line number to determine the error. For example, the message “symbol already defined” will be displayed when a duplicate variable name is found in the rack configuration and the offending variable will be clearly marked on the screen. You can view a log file by using Edit Any File

from the Commands menu, and you can print it by using the text editor print command.

After the configuration file is compiled, the selected task(s) is compiled and a log file is generated for the task to store any errors and compilation statistics. The log file created has the same name as the task and a .LOG extension. If there are errors, an object file will not be created. Refer to Appendix D for application task size limits.

The Verify procedure also generates a cross-reference file for each task selected. The cross reference has the same name as the task and an .XRF extension. The cross reference will list any undefined symbols (declared COMMON in the task, but not found in the configuration generated) and any unused symbols (defined in the rack configuration, but not used anywhere in the task cross referenced). This cross reference can be printed out. See section 8.12 for more information on the rack cross reference.

Follow the steps below to verify tasks.

Step 1. Select the task(s) you want to verify.

Step 2. Select Verify from the Task menu.

Step 3. Correct any errors indicated in the configuration .LOG file, the application task .LOG file(s), and the cross reference file.

8.9 Editing the Documentation File for a Task

Use the following procedure to edit the documentation file for the selected task. Refer to section 2.1 for more information on documentation files.

Step 1. Select the task for which you want to edit the documentation file.

Step 2. Select Document from the Task menu to access the text editor.

8.10 Transferring Task Files

You can transfer all the files for one or more selected V3.4 tasks to diskettes (or another hard drive, or another library on the same drive), or transfer all the files for an AutoMax Executive V3.4 task from diskettes (or a hard drive) into the current AutoMax directory (see section 5.1.7 for additional information). The task will automatically be added to the rack. Variables declared COMMON in the task are not added to the configuration file for the rack.

This is different from adding a task using the “Copy existing Task” option in that the destination system and rack names cannot be different than the source system and rack. Transfer is usually used to update the task files in the destination rack to match files that were edited in the source rack. It can also be used to make a backup copy of selected task files (usually on diskette) that can be transferred back to the original rack to restore task files to their original state if they were edited after the transfer.

Diskettes can be formatted as needed or, to save time, you can use empty pre-formatted diskettes. Use the following procedure to transfer tasks. Required data fields are denoted by (R) and optional data fields are denoted as (O). Some fields may contain default entries. If you do not change these fields, the default will be used.

Step 1. Select Transfer from the Task menu. The Transfer dialog box will be displayed with the current drive and library (as designated in AutoMax Setup) listed in a box on the top left.

Step 2. Enter information for the following fields:

Direction arrows - Transfer the task(s) To (left arrow) or From (right arrow) the current library. Select the appropriate option button.

Drive - Enter the source drive or the destination drive. If A or B is entered, the Executive software assumes the drive is a floppy drive; otherwise, AutoMax assumes a hard drive.

Library - Enter the library directory to copy the task(s) from or to.

Format Floppies - Select if you want to format destination diskettes.

High Capacity - Select if you want to format and/or write to high density diskettes. The appropriate diskette capacity will be displayed.

Task(s) - The Executive software will list the task(s) selected when the Transfer program was called. If you are transferring out of the current library, make certain that this field lists all of the tasks you want to transfer.

If you are transferring tasks in, you can select the tasks to be transferred. Select “Select From Tasks on disk x:” to display a Task List list box. Select each task you want to transfer and add it to the Tasks field by selecting Add to List. When you are finished, select “Done.” All the tasks you want to transfer should appear in the Tasks field.

The source code and the documentation files for the task are always transferred. Select any of the following files you want to transfer with the task:

Object files - compiled application task.

Cross-reference files - cross reference file for the rack.

Step 3. Select Cancel to return to the Task Manager without transferring a task, or OK to begin the task transfer.

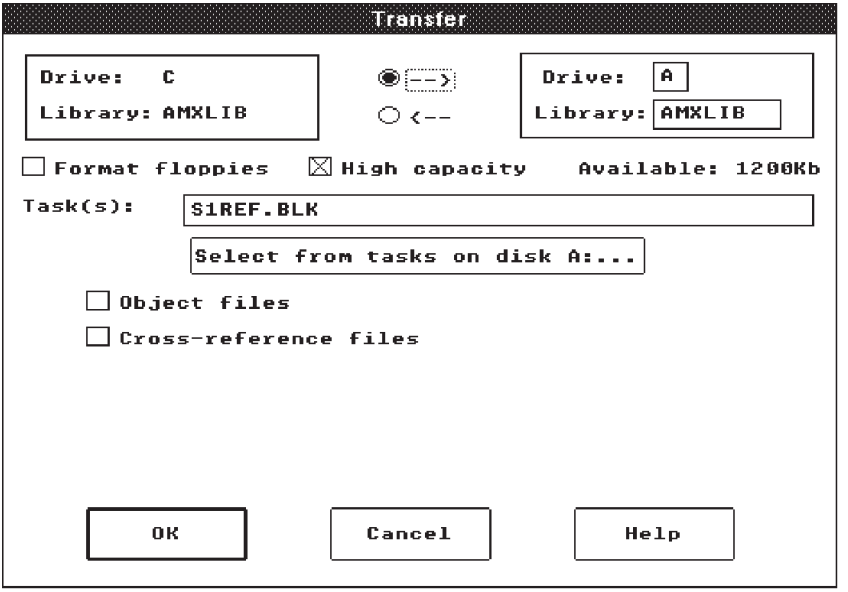


Figure 8.4 - Transfer Tasks

8.11 Printing Task Files

The Print Task command enables you to print out files relevant to one or more racks. After selecting one or more tasks from the task list, select Print from the Task menu. A dialog box with the various print options will be displayed (see the figure below).

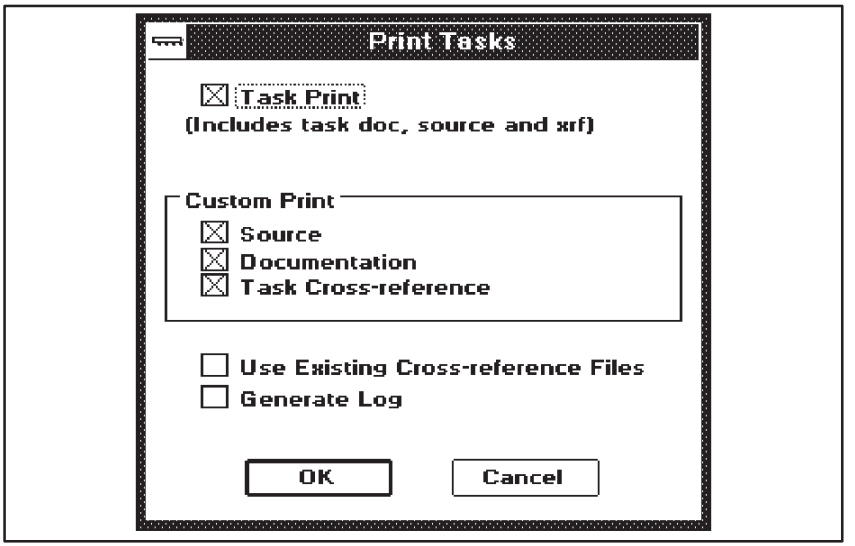


Figure 8.5 - Print Tasks Dialog Box

Selecting **Task Print** will print out the following for each of the tasks selected:

- Task title page (task name, description, engineer, slot, priority, critical, reconstructible, utility).
- Task documentation file
- Task source code
- Task cross reference file

If you de-select (un-check) Task Print, the Custom Print options will be enabled.

Select **Source** to print the source code (or ladder diagrams for PC tasks).

Select **Documentation** to print the task documentation file.

Select **Task Cross Reference** to print the cross reference file for the selected task(s).

Select **Use Existing Cross Reference Files** if you know that the cross reference files are current. Selecting this option will save time since new files will not be generated before printing. If cross reference files do not already exist, they will be generated and printed. Note that cross reference files generated by the print routine will be deleted after printing.

Select **Generate Log** to create a log file that will list any errors that may occur during the print job. The log file will exist under the rack directory and will be named _PRINT.LOG. If there are no errors, the log file will simply list the files that were printed.

8.12 Generating Task Cross Reference Files

Use the Generate Task Cross Reference command to create a cross reference file for each of the selected AutoMax and UDC application tasks. The cross reference listing will include the following for each variable used in the task: the line on which the variable is defined, the variable type (local, tunable, or reserved for the system), the hardware address (slot, drop, remote slot, register, and bit), the description assigned when the variable was configured (either the !-type comment on the application task line on which the variable is first referenced, or the 40-character element description in PC/Ladder Logic tasks), and the line cross reference. The task cross reference then lists the following errors or warnings for the task: minor compilation errors, undefined variables, unconfigured variables, local variables read but not written to, unused variables, and local variables written to but not read. AutoMax will assign cross reference files a .XRF extension. Cross reference files can then be printed by using the Print command (see section 8.11).

Step 1. Select the task(s) for which you want to create a cross reference.

Step 2. Select Generate Task Cross Reference from the Task menu.

8.13 Resequencing a Task

Resequence is used to resequence the line numbers within a task program. To resequence Control Block (BLK) and BASIC (BAS) files, follow the procedure below.

Step 1. Select the task you want to resequence.

Step 2. Select Resequence from the Task menu.

Step 3. Enter old first sequence number, old last sequence number, the new starting sequence number, and the increment. The first and last sequence numbers are inclusive.

To resequence a PC task:

Step 1. Select the task you want to edit.

Step 2. Select Edit from the Task menu to access the PC Editor.

Step 3. Press F3; then select Resequence by typing R.

Step 4. Enter old first sequence number, old last sequence num ber, the new starting sequence number, and the increment. The first and last sequence numbers are inclusive.

8.14 Selecting All Tasks

The Select All command on the Task menu highlights (selects) all the tasks in the rack. Use Select All when you want to remove, transfer, or print all of the tasks displayed.

8.15 Using Version Control Library

If your AutoMax system is using the Version Control Library facility, the command “Version Control Library” will appear on your Task menu. Refer to Appendix P for descriptions of how to use the VCL commands.

8.16 Commands Menu

The Commands menu can be used to edit non-application files that are not generated with the Executive software, to delete unneeded files, or to generate and compile the object code for the rack configuration and parameter files.

8.16.1 Editing DOS Files

Select Edit Any File from the Commands menu to edit DOS files only. Do not use this command to edit files with reserved extensions, such as task files or database files. This command displays a list box with all the available directories and files. Selecting OK invokes the text editor. Selecting Cancel returns you to the Task Manager. See 5.4.1 for additional information.

8.16.2 Purge Files

Selecting Purge from the Commands menu allows you to delete files that are no longer needed for the selected rack. All files of the selected type(s) will be deleted. This procedure will make room on the disk for other files. Source files of application tasks cannot be purged so that object files can be re-created if they are needed at a later time. Follow the procedure below to purge files.

Step 1. Select Purge from the Commands menu. A dialog box will be displayed.

Step 2. Select one or more types of files to delete:

Cross-reference (.XRF) files
Log (.LOG) files
Backup files (.~xx)
Object (.OBJ) files

Step 3. Select OK to purge or Cancel to return to the Task Manager without purging.

8.16.3 Generating the Rack Configuration and Drive Parameter Files

The Generate Configuration command (from the Commands menu) is used to generate the rack configuration object file or, for UDC modules in the rack, the drive parameter object file. When the Generate Configuration command is selected, the following options will be displayed: Generate Rack Configuration (either standard or mapping I/O locations to memory for testing) and Generate Drive Parameter Files. The next two sections describe these options.

The Transfer command from the On Line menu is used to load the rack configuration object file into the AutoMax Processor. The same command is also used to load the drive parameter object file to the UDC. See section 14.3 for information about loading the drive parameter object files.

8.16.3.1 Generating the Rack Configuration

The Generate Configuration command can be used to create the object file (.CNF) for the rack configuration. The “generated” (compiled) rack configuration must be loaded onto the Processor(s) in the rack before or at the same time application tasks are loaded. See 14.3 for information about loading the rack configuration file(s). See Appendix D for information about configuration file size limits.

When generating the rack configuration, you have the option of mapping variables to common memory for testing. If you select this option, all variables currently configured as I/O registers or I/O points will be configured as common memory locations. This allows you to load application tasks into the Processor(s) in the rack and run/monitor them without actually placing I/O modules in the rack, or without connecting the I/O modules in the rack to field devices. The variables can then be monitored through the On-Line command. Loading the configuration using this option will make the configuration take up more room on the Processor than loading without this option. Tasks that depend on hardware interrupts, i.e., that use BASIC language WAIT ON statements, will not run. Note that this option will not necessarily simulate real-world results and that

once testing is complete and you connect all field wiring to the rack, you must re-load the configuration without the option.

8.16.3.2 Generating the Drive Parameter Files

The Generate Configuration command can be used to create the drive parameter object files (.POB) for all UDCs in the rack. To create the drive parameter files, choose the Generate Drive Parameter Files option. A file named PARAMXX.POB will be created, where XX is the slot number of the UDC module. The drive parameters must be loaded onto the UDC(s) in the rack before or at the same time UDC application tasks are loaded. See 14.3 for information about loading the drive parameter files.

You can print the drive parameters for a UDC module by using the Print command from the Rack menu in the System Configurator.

8.17 On_Line! Command: Accessing Online Functions

This command is used to access the online functions. It is the same as the On_Line! command in the System Configurator. See section 5.6 for more information.

9.0 SYSTEM SECURITY

To protect the system against unauthorized online changes being made from the personal computer, AutoMax incorporates three types of security: a keyswitch on the rack Power Supply module, a password, and access levels. Note that access levels need to be considered only if multiple users will need simultaneous access to the same rack.

9.1 Keyswitch

The keyswitch on the rack Power Supply module provides security for online operations such as starting and stopping application tasks by defining a privilege level for the rack. Only online operations accessed through the ON_LINE option on the Task Manager menu are affected by the position of the keyswitch. Saving tasks from the rack, however, is permitted regardless of the keyswitch position. The keyswitch has three possible settings: “PROTECT”, “SETUP”, and “PROGRAM”. The AutoMax Executive software displays the current position of the keyswitch on the top of the screen while you are online. See below for a description of the privilege level defined by each keyswitch position.

9.1.1 PROTECT Position

If the keyswitch is in the “PROTECT” position, the user can only monitor variables and save tasks from the rack. Even if the user knows the password (see 9.2), he cannot perform any other operations.

9.1.2 SETUP Position

If the keyswitch is in the “SETUP” position, the user can monitor all variables, and save tasks from the rack only. If the USER knows the password (see 9.2), he can also modify tunable variables. He cannot perform any other operations.

9.1.3 PROGRAM Position

If the keyswitch is in the “PROGRAM” position, the user can perform all online operations from the personal computer, i.e., load, save, run, delete, and stop application tasks, and monitor and tune variables. If, however, the password (see 9.2) has not been entered previously, or is not entered when the user is prompted, the privilege level defaults to that of the “PROTECT” position.

9.2 Password and Access Levels

The password provides an additional level of security for the rack if the keyswitch on the power supply is kept in the “PROGRAM” position. Any online operations beyond what is allowed in the “SETUP” mode of the keyswitch will require the user to enter the password first. The password will time-out after 2 minutes.

Up to four users can access password-protected features in an AutoMax rack via the AutoMax network. See section 12.5 for how to connect to a rack over the network. In order to prevent conflicts, users will be granted one of the following access levels when they go online.

None - This level provides the user with monitoring capability only. The correct password has not been entered by the user.

Data Access - This level provides the user with the ability to Set/Tune/Force only common or I/O variables. Data Access will not be granted when another user has Rack Access. All users connected to a rack may have Data Access. It is the responsibility of all users to coordinate the effect of setting, tuning, or forcing on common variables and other areas of conflict.

Task Access - This level provides the user with the capabilities of Data Access plus the ability to Run/Stop/Delete/Load a task and to modify the local variables in the task using the Set/Tune/Force functions. If the user has Task access to all the tasks in a rack, he can Run/Stop/Delete/Load ALL tasks. Other network users trying to access the task will be denied access and will be shown the username of the user with privileged access. A user may have access to one or more tasks as long as no other user has access to those tasks and no other user has Rack Access.

Rack Access - This level gives the user complete control of all tasks in the rack. A user with this access level can Run/Stop/Delete/Load ALL tasks. Rack Access will not be granted if any other user has Data, Task, or Rack Access.

Users may set their level of access by using the Set Access function from the Connect menu (see 12.2). A specific level of access will be granted (if possible) when the user executes a function which requires that level of access. For example, if a user had previously entered the password and then attempts to force a local variable, Task Access will be granted to the user for the task containing the local data if no other user has access that task. Table 9.1 lists the access levels required to access password-protected features.

Data Access will be granted when the user first enters the password. The user will then be given the opportunity to acquire Rack Access. This will prevent other users from accessing the protected features in the rack. If another user has already reserved the rack, a message will be displayed on the user’s workstation that identifies the other users connected to the rack and their access levels. The user can also display a list of all other users currently logged onto the rack with their associated access levels (see 12.6).

Table 9.1 - Accessing Password-Protected Features

Rack Power Supply Key-switch Position	Any Posi-tion	PRO-TECT	SETUP			PROGRAM		
User's Access Level	**None**	**Any level**	**Data**	**Task**	**Rack**	**Data**	**Task**	**Rack**
Action								
Force Common	No	No	No	No	No	Yes	Yes	Yes
Force Local	No	No	No	No	No	No	Yes	Yes
Force I/O	No	No	No	No	No	Yes	Yes	Yes
Set Common	No	No	No	No	No	Yes	Yes	Yes
Set Local	No	No	No	No	No	No	Yes	Yes
Set I/O	No	No	No	No	No	Yes	Yes	Yes
Set/Tune Tunable	No	No	No	Yes	Yes	No	Yes	Yes
Load Normal Config	No	No	No	No	No	No	No	Yes
Load Debug Config	No	No	No	No	No	No	No	Yes
Load Single POB File	No	No	No	No	No	No	Yes[1]	Yes
Load Single Task	No	No	No	No	No	No[2]	Yes	Yes
Delete Task	No	No	No	No	No	No	Yes	Yes
Start Task	No	No	No	No	No	No	Yes	Yes
Stop Non-Critical Task	No	No	No	No	No	No	Yes	Yes
Save Task from Rack	No	Yes[3]	No	Yes	Yes	No	Yes	Yes
Load All POB Files	No	No	No	No	No	No	Yes	Yes
Load All	No	No	No	No	No	No	No	Yes
Delete All Tasks	No	No	No	No	No	No	Yes[4]	Yes
Start All Tasks	No	No	No	No	No	No	Yes[4]	Yes
Stop All Tasks	No	No	No	No	No	No	Yes[4]	Yes
Load AutoMax OS	No	No	No	No	No	No	No	Yes
Load Single UDC OS	No	No	No	No	No	No	Yes[1]	Yes
Load All UDC OS	No	No	No	No	No	No	Yes[1]	Yes
Modify PC Task	No	No	No	No	No	No	Yes	Yes
Auto Run	No	No	No	No	No	Yes	Yes	Yes
Monitor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Notes: A user with privileged access may relinquish this privilege by timing out after two minutes offline, or by selecting to relinquish access online. Functions not listed in the table do not require privileged access.

(notes continued on next page)

(Continued)

1. Must have TASK Access to both tasks in UDC.
2. A single task may be loaded with only DATA Access if the task does not already exist on the Processor or UDC module.
3. Requires TASK or RACK Access.
4. Must have TASK Access to all tasks.

As supplied by Reliance, the AutoMax Executive software requires the password "AUTOMAX". To change the password, you must use one of the two methods described below in 9.2.1 and 9.2.2.

The two utilities which can be used to change the password are not installed on the hard disk along with the AutoMax executive. They reside on disk 1 of the AutoMax software. In order to use either method of changing the password, the user must have access to the original AutoMax installation disks, or the backup copy of those disks.

9.2.1 Changing the Password in the AutoMax Operating System File

This method of changing the password alters the AutoMax operating system file *.OS that is stored with the AutoMax Executive software in subdirectory <DRV>:\RPE\AMX3 on the hard disk, and that is also loaded onto the rack (see 5.4.3 for information about loading the operating system). After changing the password using this method, you need to load, or re-load, the operating system containing the new password into each rack. Note that if you load the Ethernet version of the operating system, you can still use this method of changing the password, i.e., follow the exact same directions below.

Note that you will need to enter the current (old) password in order to load the operating system onto the rack. Follow the steps below to change the password:

1. Run the AutoMax Executive software by typing AUTOMAX3<CR>.
2. Click the Control menu in the upper left hand corner and select Switch To in order to access the Program Manager. Select the DOS icon.
3. Insert disk 1 of the original or backup disks of the AutoMax Executive software into drive A:
4. Type A:PWOS<CR>
5. When the system prompts you for the new password, enter the password you have chosen. You may enter a <CR> as the password. The operator will then be required to enter a <CR> in order to perform the online operations that require the password.

   The operating system file with the old password is automatically re-named *.~OS and the operating system with the new password is named *.OS.
6. When you are finished and back at the DOS prompt, type EXIT<CR>. This will return you to the Program Manager. Use Switch To to return to the System Configurator.

7. Load (or re-load, if applicable) the operating system into each rack using the procedure outlined in 5.4.3. When you re-load the operating system, you will be prompted for the password. This password is the current password for the rack, NOT the new password in the operating system that will be loaded onto the rack.

9.2.2 Changing the Password in a Rack

The method outlined below changes the password directly in the AutoMax Processor module without making any changes in the AutoMax Executive software. In order to use this method, the operating system must already be loaded onto the Processor (see 5.4.3). The Processor does not have to be stopped and the operating system does not have to be re-loaded when you use this method to change the password. In addition, to use this method, you do not need to know the current password to change it. Follow the steps below to change the password:

1. Run the AutoMax Executive software by typing AUTOMAX3<CR>.
2. Click the Control menu in the upper left hand corner and select Switch To in order to access the Program Manager. Select the DOS icon.
3. Insert disk 1 of the original or backup disks of the AutoMax Executive software into drive A:
4. Type A:PWRACK <CR>
5. When the system prompts you for the new password, enter the password you have chosen. You may enter a <CR> for the password. The user will then be required to enter a <CR> when performing online menu operations that require the password.
6. When you are finished and back at the DOS prompt, type EXIT<CR>. This will return you to the Program Manager. Minimize the Program Manager to return to the System Configurator.

Note that with the above method for changing the password, the edited password file resides only in the AutoMax Processor module(s). The password can not be edited or read from the Processor module(s).

10.0 AutoMax PROCESSOR OVERVIEW

The AutoMax Processor module plugs into the backplane of an AutoMax rack and executes application programs which, in turn, control other AutoMax or DCS modules in the system. The Processor modules have the following memory/speed configurations:

Model	CPU Speed	Memory
M/N 57C430A AutoMax 6010 Processor module	8 mHz	256K Parity RAM
M/N 57C431 AutoMax 6011 Processor module	8 mHz	512K Parity RAM
M/N 57C435 AutoMax 7010 Processor module	25 mHz	512K Parity RAM

Each Processor contains 32K of EPROM for board-level diagnostics and boot software. The operating system, which oversees the operation of the CPU and the execution of application tasks, is provided in two versions. The version that includes the Ethernet functions will occupy approximately 134K of RAM, leaving 122K available for application tasks on the M/N 57C430A Processor. The standard operating system will occupy approximately 119K of RAM, leaving 137K available for application tasks on the M/N 57C430A Processor. The M/N 57C431 and M/N 57C435 Processors have 300K available for application tasks regardless of which operating system is being used. Both versions of the operating system are included with the AutoMax Executive software (see section 5.4.3 for loading the operating system).

Processor modules come equipped with on-board battery back-up to protect against power failures. External battery back-up is therefore not required for racks containing a single Processor module. For racks containing multiple Processor modules, however, external battery back-up may be required to protect the Common Memory module against power failures. If you are using Common Memory module M/N 57C413 or earlier, external battery back-up is required. M/N 57C413B and later have on-board battery back-up. Although this section describes the Processor module in some detail, for more specific information, refer to instruction manual J-3650.

10.1 Single Processor Module in a Rack

A Processor can occupy any slot in the rack from 0-4 (refer to figure 10.1). With the standard operating system loaded, a single M/N 57C430A Processor module in a rack makes available approximately 137K of memory for application tasks. A single M/N 57C431 or 57C435 Processor module makes 300K available for application tasks.

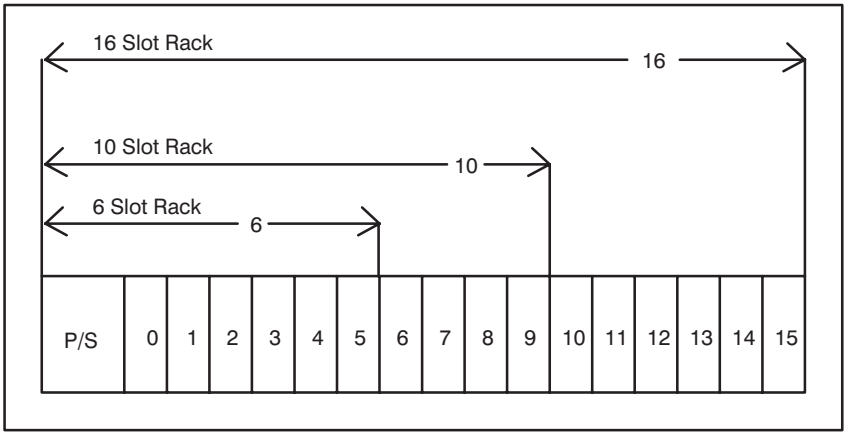


Figure 10.1 - Rack Slot Numbers

10.2 Additional Data Storage in a Rack Containing a Single Processor Module

A Common Memory module M/N 57C413A or 57C423 can be used to extend the amount of data storage available in a rack containing a single Processor module. In this configuration, the M/N 57C413 or 57C423 Common Memory module makes available an additional 128K of memory to use for storing common variables. You cannot use the Common Memory module to store application tasks or local variables. See instruction manual J-3675 for more information on common and local variables.

10.3 Multiple Processor Modules in a Rack

Additional Processor modules increase the processing capability and the total memory available in any one rack for application tasks. **Multiple Processor modules in a rack require the use of a Common Memory module in slot 0,** where it serves as a bus arbiter and stores data common to all the Processor modules. The Common Memory module can be used to oversee communications for up to four Processor modules, the maximum per rack. The Processor modules can occupy slots 1-4 only. In a rack containing multiple Processors, slot 1 must either contain a Processor module or be empty.

In this configuration, the Common Memory module, rather than a Processor module, also stores the configuration data. See J-3636 for more information about the Common Memory module.

If you are using M/N 57C430A or 57C431 and M/N 57C435 Processors in the same rack, make sure the left-most Processor is a M/N 57C435 Processor. This will allow all communication between the AutoMax rack and the personal computer to occur at 19200 baud.

10.4 Battery Back-Up

The contents of AutoMax Processor RAM (read/write memory) are preserved through power failures by the on-board battery back-up. As long as the LED labeled “BAT. OK” on the front of the Processor module is on, the battery is functional and 115 VAC is available to the power supply in the rack. Should the system lose power, the on-board battery can supply power to the Processor module for a minimum of 42 days. Note that the battery backup is designed to maintain the contents of RAM only. It is not a source of un-interruptible power.

If you expect power to be off for long intervals during initial start-up and debugging, you should disconnect the battery backup on the Processor module and make use of the super-capacitor also on board each Processor module. Typically, the super-capacitor is capable of retaining memory for 10 hours at a time with no battery present. This procedure will avoid draining the Processor’s battery back-up.

Reserved Battery Status Variables

Five pre-assigned variables are available for use in all application tasks to test the status of the on-board battery of AutoMax Processor and Common Memory modules. These common boolean variables will have the value 1 if the battery is functional and 0 if the battery is not functional. The variables are named according to the Processor whose battery is being tested. BATTERYSTATUS0@ is used for the Processor or Common Memory module in slot 0. BATTERYSTATUS1@ is used for the Processor in slot 1, etc. up to slot 4.

10.5 Module Watchdog Circuity

Each Processor module has a local watchdog timer which must be reset by the operating system within a specified interval or the Processor will execute a STOP ALL and all I/O modules in the local rack will be reset (initialized to 0, FALSE, or OFF). The I/O modules in the rack will also be reset if you remove a Processor module from a single-Processor configuration. All UDC tasks in the rack will also be stopped, and most UDC registers will be reset. See 10.12 for more information.

In a multi-Processor configuration, there is a system watchdog timer located on the Common Memory module in slot 0 in addition to the watchdog on each individual Processor module. If the system watchdog is allowed to expire, the Common Memory module will generate an interrupt, and one of the Processors in the rack will issue a STOP ALL and reset the I/O modules in the rack. All Processor modules in the rack will then shut down.

Each UDC module also has a local watchdog timer. If the watchdog timer on a UDC module expires, the UDC module will generate an interrupt, and one of the Processors in the rack will issue a STOP ALL.

10.6 Processor Module Serial Ports

There are two RS-232 serial ports on the front of each Processor module. They are accessed through DB-25 connectors on the faceplate. The upper port, labeled “PROGRAMMER/PORT B”, is reserved for connection to the personal computer only. When there are multiple Processor modules in a rack, only the leftmost “PROGRAMMER/PORT B” slot is reserved. All remaining ports on all Processor modules in the rack can be used by application tasks running on the respective Processor modules.

The user ports are accessed using the OPEN statement (OPEN “PORTA” or OPEN “PORTB”) in BASIC tasks. Refer to J-3675 for more information on the OPEN statement. Refer to Appendix A for a description of the pins on user ports. Unless otherwise programmed by an application task, the default characteristics of the user ports are:

9600 baud
8 bit characters
1 stop bit
no parity
echo on
Xon/Xoff handshake enabled
ignore modem control
not hardcopy device

10.7 Status Indicators

The Processor module has four status indicators on the faceplate: a green LED labeled “OK”, an indicator light labeled “BAT. OK” for the on-board battery back-up and presence of 115 VAC power, and two seven-segment LEDs used to display status and error codes. See 10.4 for more information on the on-board battery back-up.

The “OK” indicator is controlled by the local hardware watchdog timer on the Processor. It is on when the timer has been reset within the timer interval and the Processor module is operating normally. See 10.5 for more information about the watchdog timer.

See 10.10-10.12 for more information on status and error codes that may be displayed on the seven-segment LEDs.

10.8 Power-Up Initialization

Whenever power is cycled, each Processor and UDC module performs diagnostics to detect any malfunctions on the module. In a rack with multiple Processors, the leftmost Processor will perform diagnostics on the Common Memory module.

While the diagnostics are being performed, various status codes are displayed on the Processor, indicating the particular diagnostic in progress. Most of these codes are displayed so briefly that they are not actually visible. Should a failure be detected on a Processor or Common Memory module, the Processor will be shut down and the “OK” indicator turned off. The status code indicating which diagnostic failed will remain on the Processor’s display. See Appendix B for a list of status and error codes. A failure on a UDC module will cause the “OK” indicator to be turned off.

After the diagnostics are complete, a checksum of a portion of the contents of memory is verified against a checksum which was stored in memory when a loss of AC power was detected. This determines whether the system performs a re-start or a cold-start when power is turned on. See 10.13 and 10.14 for more information on cold starts and system re-starts.

10.9 Run-Time Diagnostics

The Processor module performs real-time checking of all data paths on the module by means of a parity test. The Processor contains a memory management unit (MMU) that prevents errors such as writing to locations that are read-only.

Should a failure be detected, a fault code will be displayed and the Processor shut down. See instruction manual J-3650 for more information on run-time diagnostics.

10.10 Status Codes

Status Codes are those codes displayed on the seven-segment LEDs on the Processor module faceplate while the “OK” indicator is on.

Status codes simply indicate that a particular operation is occurring or that a particular condition exists. They do not cause the Processor to shut down. Status codes may or may not be cleared when the condition they indicated no longer exists. See Appendix B for a list of status codes.

10.11 Shut-Down Faults

Serious hardware malfunctions that cause the Processor to shut down and the “OK” indicator to be turned off are called shut-down faults. When possible, the cause of the shut-down is indicated by an error code on the seven segment LEDs. See Appendix B for a list of status and error codes.

Once a Processor module has shut down, it will not execute any instructions or respond to commands from the personal computer until it is reset by cycling power.

10.12 Stop-All Faults

In AutoMax systems, both a Stop-All command and a Stop-All fault have the same result. A Stop-All command can be issued from the online menu of the Programming Executive software. A Stop-All fault occurs when there is a serious error either in an application task in the rack (e.g., invalid BASIC language EVENT statement), or when there is a serious error in an AutoMax Processor or Universal Drive Controller module. See Appendix B for a list of status and error codes.

Both a Stop-All fault and a Stop-All command will result in all application tasks in the rack being stopped. The common clock signal on the rack backplane (CCLK) will also be disabled. A Stop-All fault will result in an error code display on the faceplate of the AutoMax Processor that was running the task that caused the Stop-All. Neither the operating systems or the application tasks on

AutoMax Processors and UDC modules will be deleted by Stop-All faults or commands.

The effect of a Stop All on data in the rack depends on the data type. Local tunable data in both AutoMax and UDC application tasks is always retained. Local data is retained for AutoMax tasks, but not for UDC tasks. Common memory data, which can be defined in the Variable Configurator for AutoMax Processors only, is maintained when it is configured as non-volatile. Otherwise, it is reset to 0. For I/O data, inputs are retained and continue to be updated, while outputs are reset (set to 0 or off). Note that the UDC dual port memory is treated like I/O data in the system.

AutoMax Processors will retain the last values of all local variables and non-volatile common memory variables. All I/O input values are retained and continue to be updated. All I/O outputs in the rack and in any remote I/O racks will be reset.

UDC modules will retain their parameter configuration data, UDC test switch information, D/A setup configuration, local tunable variables, and the following input data: feedback registers and UDC task error log information.

UDC modules will NOT retain local variables and data found in the following registers, which are considered outputs: command registers, application registers, the ISCR (interrupt status and control register), scans per interrupt register, and scans per interrupt counter register. The UDC-PMI communication status registers will not be retained.

For Distributed Power Drive applications, the PMI Processor connected to UDC modules will react to a Stop All as follows. All I/O in the PMI rack, including the rail I/O and I/O connected to the Resolver and Drive I/O module, is reset. Because a Stop All causes the CCLK signal being used to synchronize UDC and PMI Processor communication to be turned off, the UDC and PMI will become unsynchronized. The PMI Processor reacts to all serious synchronization problems in the same way: armature and field current reference will be set to 0, and the PMI Processor will continue commanding 0 current until it senses discontinuous conduction. At this point, the PMI Processor will turn off the M- contactor output signal on the Resolver and Drive I/O module in the PMI rack.

10.13 Forcing a Cold Start/Deleting Operating Systems

At times it may be necessary to cold start the Processor(s) to erase the contents of RAM. Because of the on-board battery back-up, you cannot erase the contents of RAM simply by cycling power. To force a cold start, perform the steps below.

Before proceeding, note carefully that this operation will erase the contents of RAM for all Processors in the rack and that it will require re-loading the AutoMax OS, the rack configuration and all application tasks. Make certain that you have an accurate backup of all tasks in the rack, including utility tasks. See 14.3.1 for more information on utility tasks.

Read all directions carefully before starting the procedure.

1. Turn off power to the system. All power to the rack, as well as all power leading to modules in the rack, must be off.

2. Take any Processor module in the rack out of its current slot and insert it into slot 5, 6 or 7. If there are modules in these slots already, take one out and set aside for the moment. Then insert the Processor into the empty slot.
3. Turn on power to the rack.
4. Turn off power to the rack.
5. Move the Processor back to its original slot and re-insert any other module taken out in step 2 above.
6. Turn on power to the rack. The leftmost Processor will execute its power-up diagnostics and then display “L” “O” on its LEDs.
7. Re-load the AuoMax OS, rack configuration, and all application tasks.

10.14 Stopping and Re-Starting AutoMax and UDC Tasks in the Rack

WARNING

DEPENDING ON THE APPLICATION, STOPPING AN INDIVIDUAL TASK MAY RESULT IN LOSS OF CONTROL OF THE APPLICATION PROCESS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. IT IS RECOMMENDED THAT STOP-ALL BE USED TO STOP TASKS NORMALLY. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

THE STOP-ALL FUNCTION STOPS THE APPLICATION PROGRAMS IN A RACK. IT IS THE RESPONSIBILITY OF THE USER TO ENSURE THAT THE APPLICATION PROCESS STOPS IN A SAFE MANNER WHEN THE APPLICATION PROGRAMS STOP. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

THE STOP-ALL FUNCTION MAY CAUSE THE OUTPUTS TO CHANGE STATE, RESULTING IN MACHINE MOVEMENT. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

The following is a description of what happens to a rack when tasks are stopped under different circumstances. It also describes how tasks can be re-started under those circumstances. Refer to section 10.15 and 12.4 for more information.

1. You stop a task from the personal computer using the STOP command described in 16.1. The outputs controlled by this task will remain at their last state.

The task is stopped and will not go into run again until you put it into run using the RUN command described in 15.1.

2. You stop all tasks in the rack to which you are connected, either directly or over a network, using the STOP-ALL command described in 16.2.

   All tasks in the rack are stopped and all I/O modules are cleared, i.e., set to 0, false, or off.

   Application tasks will not go into run again until you put them all into run using the RUN-ALL command described in 15.2. You can also put the tasks into run individually using the RUN command described in 15.1.

3. A Stop-All fault occurs.

   All tasks in the rack are stopped and all I/O modules are cleared. The Processor module detecting the fault will display an error code. See 10.12 for more information about stop-all faults.

   Before you run tasks again, you should first clear the error log described in 19.0. Application tasks will not go into run again until you use the RUN or RUN-ALL commands described in 15.1 and 15.2, respectively.

4. You cycle power or a power failure occurs.

   All tasks in the rack are stopped and all I/O modules are cleared. When power is returned to the system, a re-initialization occurs.

   Application tasks in the rack can be put into run in two ways after power comes back on: manually, using the RUN or RUN-ALL commands described in 15.1 and 15.2, respectively; or automatically by the system if AUTO-RUN was enabled at the time power went off.

   AUTO-RUN puts into run only those application tasks that were running when power went off. It will not put any tasks into run if any errors occurred prior to power going off, or if any errors occur during the system re-start.

10.15 System Re-Initialization

System re-initialization refers to a condition that occurs in a rack that contains at least one Processor module. It does not necessarily affect other racks on networks connected to the rack through 57C404. 57C404A, or 57C404B modules.

A system re-initialization will occur under the following conditions:

- when the system powers-up after power to the system is turned off either by cycling power or through a power failure,
- after the occurrence of a STOP-ALL fault,
- after you issue a STOP-ALL command

The following steps occur during a system re-initialization:

1. The following codes are cleared:

   drive fault codes “81” through “86”
   status codes “d0” through “dF”
   configuration task error codes “E0” through “EF”

   Any other fault codes are re-displayed.

2. An application configuration check is performed. If the application configuration check determines that an I/O module is missing or not functioning, an error code is displayed.
3. Forced outputs are set to their forced values.

WARNING

VARIABLES AND OUTPUTS WHICH ARE FORCED BEFORE AC POWER IS LOST WILL REMAIN FORCED WHEN AC POWER IS RESTORED. SHOULD AC POWER BE LOST WHILE VARIABLES ARE FORCED, THE USER MUST ENSURE THAT UNEXPECTED MACHINE MOVEMENT DOES NOT OCCUR WHEN AC POWER IS RESTORED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

4. Volatile common variables are cleared, i.e., initialized to 0, false, or off, if they are not forced. The values of local variables, non-volatile common variables, and forced volatile common variables are retained. All I/O points are cleared.
5. The application tasks are “built,” i.e., the system data structures are created.
6. The application tasks are installed. The status code “d0” is displayed during the installation because this may take a significant amount of time, depending on the size and number of tasks to be installed. The Processor attempts to read all I/O locations to verify that they are still there.
7. Application tasks may be re-started if the re-initialization was the result of power being turned off and AUTO-RUN is enabled. Otherwise, application tasks are not automatically re-started. See 12.4 for more information on AUTO-RUN.

11.0 AutoMax ON LINE MENU

The ON LINE! command in the System Configurator and the Task Manager applications allows you to select the ON LINE menu shown in figure 11.1. Options on this menu are used to load, run, stop, monitor, and modify application tasks in the rack and to connect to other racks on the network(s). In addition, this menu provides access to the error log kept for each AutoMax Processor or UDC module running tasks.

Before attempting to use any of the options in the ON LINE menu, you must load the operating system onto the Processor modules and UDC modules in the rack if you have not already done so. Refer to 5.4.3 for directions on loading the operating system.

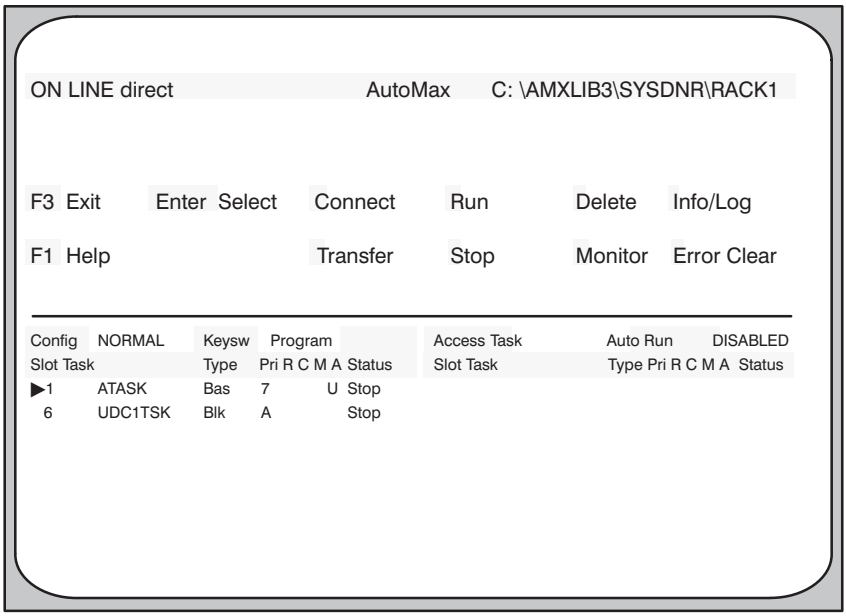


Figure 11.1 - ON LINE Menu

When you select ON LINE you will be prompted to connect the cable to the leftmost Processor in the rack and then type “Enter” or <CR> if the cable is not already connected. If you have already connected the cable, simply enter a <CR> to go online. If you have not connected the cable, refer to the directions in 3.4. Once you are online, you will see the menu shown in figure 11.1.

When you first go into the ON LINE menu, the fields in the status line displayed on the screen directly beneath the horizontal line will be empty if there are no tasks loaded on the rack at this time. A sample display when there is one task running in the rack is described in 11.3 and shown in figure 11.2.

Use F3 to exit the ON LINE menu and return to the System Configurator or Task Manager application.

11.1 ON LINE Menu Options

Online operations are not run in the Windows environment. Selections are made using the keyboard exclusively. You use the first letter of each option (shown in reverse video) instead of function keys to select the option. The default DISK:\LIBRARY\SYSTEM\RACK, i.e., the path, is displayed in the upper right-hand corner of the screen.

The “Enter (Select)” option stands for the ENTER key. Certain options will require that an application task be selected before you can choose the option. The Enter key will select the task on the screen preceded by a “>” symbol. This symbol, in turn, is controlled by the arrow keys.

In addition, the “Enter” key is often used to terminate entry of required information in fields and move to the next field.

ON LINE menu options are described in the sections below as follows:

Connect	(12.0)
Info/Log	(13.0)
Transfer	(14.0)
Run	(15.0)
Stop	(16.0)
Delete	(17.0)
Monitor	(18.0)
Error Clear	(19.0)

11.2 ON LINE Menu Security

The options available to you when you are online depend on the position of the keyswitch and whether you enter the correct password. See 12.2 for directions on entering the password. See section 9 for more information about the keyswitch and system security.

11.3 ON LINE Menu Display

The AutoMax Executive maintains a real-time display of the status of up to 32 tasks in the rack when you are online. Figure 11.2 shows the ON LINE menu as it appears when there is one task running in the rack.

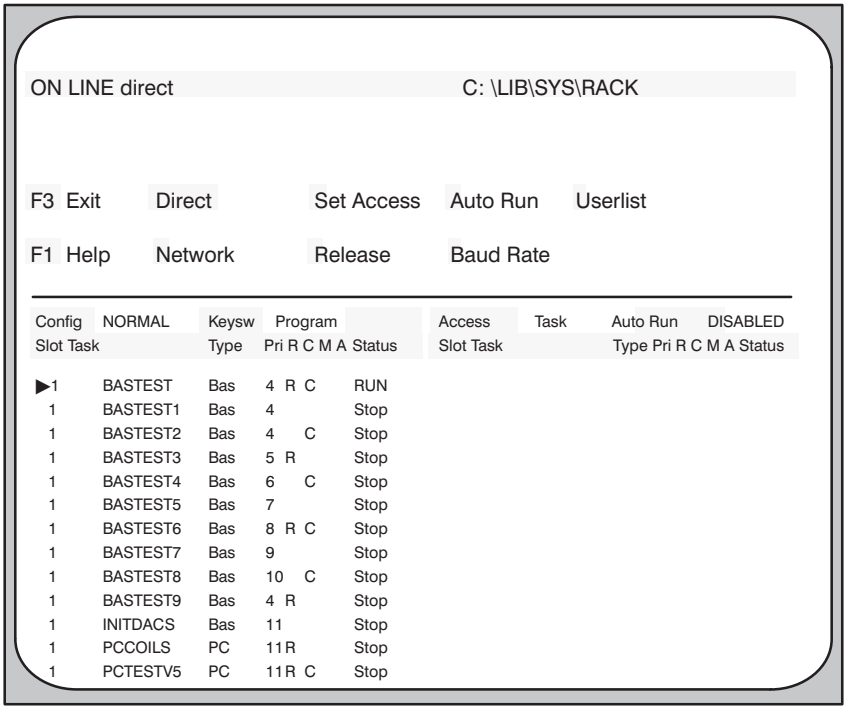


Figure 11.2 -ON LINE Menu Display with Task Running

The top status line displays general information about the rack. The “Config” field shows whether the NORMAL or DEBUG version of the rack configuration has been loaded to the rack. If the rack configuration has not been loaded, “NONE” wll be displayed. The keyswitch position, Program in this case, is shown in the “Keysw” field. The “Forced” field, when shown, indicates whether there are any forced variables in the rack. See 18.5 for more information on forcing.

The user’s access level is displayed in the field “Access” as None, Data, Rack, Locked, or, as shown here, “Task”. “Auto Run” is either “enabled” or as shown here, “Disabled”. See 12.4 for more information about AUTO-RUN.

The second status line displays specific information about tasks in the rack. “Slot” denotes the AutoMax Processor or UDC module on which the task is running. “Task” is the name of the task and “Type” denotes the programming language (.BLK, .BAS, or .PC).

“Pri” is the task’s priority as defined in the configuration for the rack. The priority ranges from 4 (highest) to 11 (lowest). UDC tasks do not display a priority. “A” is displayed for Drive A tasks; “B” is displayed for Drive B tasks. See Appendix C for more information on task priority. The field “R” will show an R if the task is reconstructible. See 8.1.1 for more information about the /Reconstruct option used in compiling tasks.

Field “C” will show a C if the task has been designated as CRITICAL in the configuration for the rack. CRITICAL tasks are those that cannot be stopped individually, i.e., all tasks must be stopped in order for the task in question to be stopped.

An M in field “M” indicates that the task has been modified, but has not been saved from the Processor module. This field is used only when tuneables have been changed either by the operator, or for UDC tasks, by the PMI operating system, or when Ladder Logic tasks have been modified. For more information about saving tasks from the Processor module, see section 14.4.

Field “A” will show a “U” if the user has task access, or “O” if another user has task access to the task. For single user (non-network) applications, this field will be blank or will display “U” if the user has edited the task.

Finally, “Status” shows whether each task loaded in the rack is currently running (RUN) or stopped (STOP), and whether there has been an error (ERROR) in each task.

12.0 ON LINE MENU: CONNECT

The Connect menu is used to change the baud rate, enter the password, release the password, set or release access, and enable/disable AUTO RUN for the rack. You can also display a list of users. These options are described below. To select the Connect menu, enter “C” for at the ON LINE menu shown in figure 11.1. The resulting menu is shown in figure 12.1.

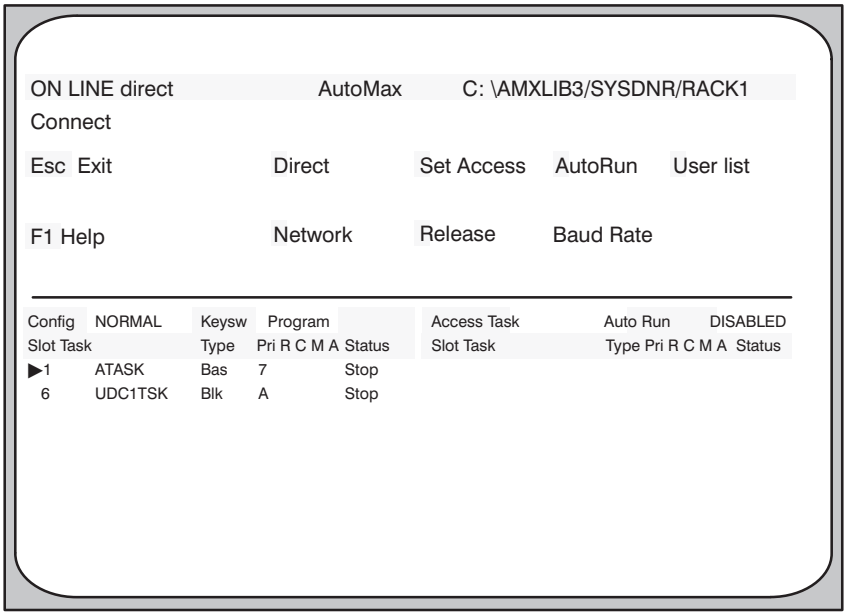


Figure 12.1 - CONNECT Menu

12.1 Changing the Baud Rate

Typing “B” for “Baud Rate” from the Connect menu shown in figure 12.1 allows you to increment or decrement the default baud rate used for communication with the rack. For the M/N 57C435 Processor, the default is 19200 baud. For the M/N 57C430A and 57C431, the default is 9600 baud. Use the up and down arrow keys to increment and decrement the baud rate displayed. The allowable baud rates are 1200, 2400, 4800, 9600, and 19200 baud.

12.2 Setting Access

Typing “S” for Set Access from the Connect menu shown in figure 12.1 allows you to set various levels of access to the tasks in the rack. If the keyswitch on the rack power supply is in any position other than PROGRAM, setting any level of access has no effect because the keyswitch has already set a lower level of privileged access. See section 9.2 for more information on security access levels.

If the keyswitch in not in the PROGRAM position, or if you do not have the required access level, the options available to you from the ON LINE menu shown in figure 11.1 are limited to monitoring variables.

If the keyswitch is in the PROGRAM position and you try to load, run or stop tasks, set or force variables, or do online editing of tasks without first entering the password, you will be prompted for the password. If you do not enter the password, you will be unable to execute these functions.

Entering the password gives you DATA access, which limits you to modifying COMMON variables. When you try to execute a function which requires a higher level of access, AutoMax will automatically grant the required level of access if it is available. Table 12.1 shows the levels of access available to you based on your current level of access and the access levels of other users connected to the rack. DATA access and RACK access are set by executing the Set Access function with no task selected. TASK access is set by executing the Set Access function with a specific task selected. For example, if you have no access when you select the Set Access function, and you have no task selected, AutoMax will request the password. If you enter the password and no other user has access to the rack, you will be asked if you want RACK access. If you respond “Y” (Yes), no other user will be able to obtain access to the tasks in the rack. If you respond “N” (No), AutoMax will ask you if you want access to all tasks.

Table 12.1 - Setting Access

User’s Present Access Level	Other Users’ Access Level	Access Levels Available to User
None	None	DATA, single task, all tasks, RACK
	DATA	DATA, single task
	TASK	DATA, single task
	RACK	None
DATA	None	single task, all tasks, RACK
	DATA	single task, all tasks
	TASK	single task
TASK	None	single task, all tasks, RACK
	DATA	single task, all tasks
	TASK	single task

The Programming Executive software is shipped from the factory with the password “AUTOMAX”. You can change the password using either of the two password utilities described in section 9.2.

12.3 Releasing Access

Typing “R” for Release Access from the Connect menu shown in figure 12.1 allows you to release various levels of access to the rack. If you have either TASK or RACK access, and you select Release Access without first selecting a task, AutoMax will ask you if you want to release your present access level as well as DATA access. If you presently have DATA access, it will be released without a prompt from AutoMax. If you selected a task before selecting Release Access, access will be released for the selected task only.

When all levels of access have been released, PROGRAM mode is effectively disabled even though the keyswitch on the rack power supply is still in the PROGRAM position.

12.4 Setting AUTO-RUN

WARNING

BEFORE TURNING ON POWER TO THE SYSTEM, YOU MUST MAKE CERTAIN YOU UNDERSTAND THE RESULTS OF ALL APPLICATION TASKS. OUTPUTS MAY CHANGE STATE, RESULTING IN MACHINE MOVEMENT. FAILURE TO OBSERVE THIS PRECAUTION MAY RESULT IN BODILY INJURY OR DAMAGE TO EQUIPMENT.

Entering “A” for “AUTO-RUN” at the Connect menu allows you to enable/disable AUTO-RUN as long as the password is entered and the keyswitch is in PROGRAM. When AUTO-RUN is disabled, all tasks will be in STOP mode on power up. When AUTO-RUN is enabled, all tasks that were in RUN mode before power to the rack was turned off will go back into RUN when power is turned back on.

By default, AUTO-RUN will be disabled when the AutoMax operating system is loaded to the rack. It will remain disabled until you enable it. After AUTO-RUN is enabled, it will remain enabled until you disable it. It will not be disabled when you enter the password, change access level, or set or force variables.

You can define a common variable, AUTORUNSTATUS@, that will indicate whether AUTORUN is enabled for the rack. See 7.2.2 for more information.

The following conditions must be met for AUTO-RUN to be initiated:

1. AUTO-RUN must be enabled through the ON LINE PROGRAMMING option on the AutoMax MAIN menu.
2. The system re-initialization described in 10.14 must occur with no errors.
3. Each task specified in the configuration for the rack except for utility tasks is present and installed on the correct AutoMax Processor or UDC module, and was running when power was cycled.

See 12.4.1 for more information about tasks that will not be put into run with AUTO-RUN even if conditions 1-3 above are met.

If there is any discrepancy between the information in the configuration for the rack and an actual application task, the fault code “E” “0” is displayed on the AutoMax Processor and no application tasks are re-started. In a multiple Processor configuration, this check is performed by one Processor only. The particular Processor on which the fault code appears bears no relation to the location of the problem.

Tasks are re-started in order of priority from highest to lowest on each Processor module. Tasks with the same priority are started in alphabetical order. Tasks on different Processors are not synchronized. In addition, unless the programmer uses interrupts to synchronize them, tasks on UDC modules are not synchronized with tasks on AutoMax Processors.

Task execution always begins from the first statement in the task, regardless of the point at which the task stopped. Tasks on different UDC modules will be re-started independently of each other. The drive A task is always executed first, followed by the drive B task. UDC task execution always begins from the first statement in the task, regardless of the point at which the task may have been stopped. After CCLK is enabled, the starting of tasks on all UDC modules in the rack will be coordinated.

12.4.1 Application Tasks Not Re-Started with AUTO-RUN

The following tasks are not re-started automatically when the system powers up and AUTO-RUN is enabled.

1. Any task that was was not in run when power was turned off. You must put the task into run manually. See 15.0 for more information about running tasks.
2. Any task that was stopped by a STOP or STOP-ALL command issued from the personal computer. You must put the task(s) into run manually. See 15.0 for more information about running tasks.
3. Any task that was stopped by the occurrence of a STOP-ALL fault described in 10.12. You should first clear any error codes from the error log or from Processor LEDs and then put the task into run manually. See 19.0 for more information about clearing the error log or LEDs and 15.0 for more information about running tasks.

12.4.2 Application Task AUTO-RUN and Memory Fragmentation

Occasionally, there may be insufficient memory in the Processor(s) after a system re-start to accommodate application tasks that were initially loaded (recall that the tasks must be “built” again) due to memory fragmentation. Since system re-starts are inevitable, after loading the rack configuration and application tasks, you should verify that there will be sufficient memory should a re-start occur. You can do this by issuing a STOP-ALL command from the personal computer or by cycling power and then checking that all application tasks are present and installed

12.5 Connecting to a Network or a Local Rack

You can communicate with an AutoMax rack by using either an RS-232 serial connection to a Processor in the rack or by using a PC Link module (M/N 57C445). You can choose which one of these methods you want to use for communication to AutoMax racks (see section 5.5.3 for instructions).

When using a serial connection, communication between the personal computer running the AutoMax Executive software and a given rack in the system can take place either directly with the local rack or over a network through the Network Communication module (M/N 57C404A). The default status is direct communication with the rack that is physically connected to the personal computer. See 12.5.1 and 12.5.2 for more information on direct and network communication.

When using a PC Link module, communication can be established with any rack on the network through the coaxial cable that is used by the network. Direct communication with the local rack is not an option when using the PC Link module. See 12.5.2 for more information on network communication.

Note that you do not need to establish a network connection through the ON LINE menu to load an operating system over a network. The loading procedure is always performed through the Commands menu in the System Configurator.

12.5.1 Direct Communication with the Local Rack

Selecting “D” for “Direct” from the ON LINE CONNECT menu will cause the Executive software to send all communication to the rack to which the personal computer is directly connected. The word “direct” will be visible on the top line of most ON LINE menu screens. Direct connection is the default status.

12.5.2 Network Communication

Selecting “N” for “Network” from the ON LINE CONNECT menu will cause the Executive software to send all communication to the network rack (drop) specified. Until you change the type of connection, either by using “N” for “Network” to connect to another network rack or by using “D” for “Direct,” all communication will take place with the specified rack.

If you are using a PC Link module, the Executive software will display the screen shown in figure 12.3. You can skip to the description that follows figure 12.3.

If you are using serial communication, the Executive software will display the screen shown in figure 12.2.

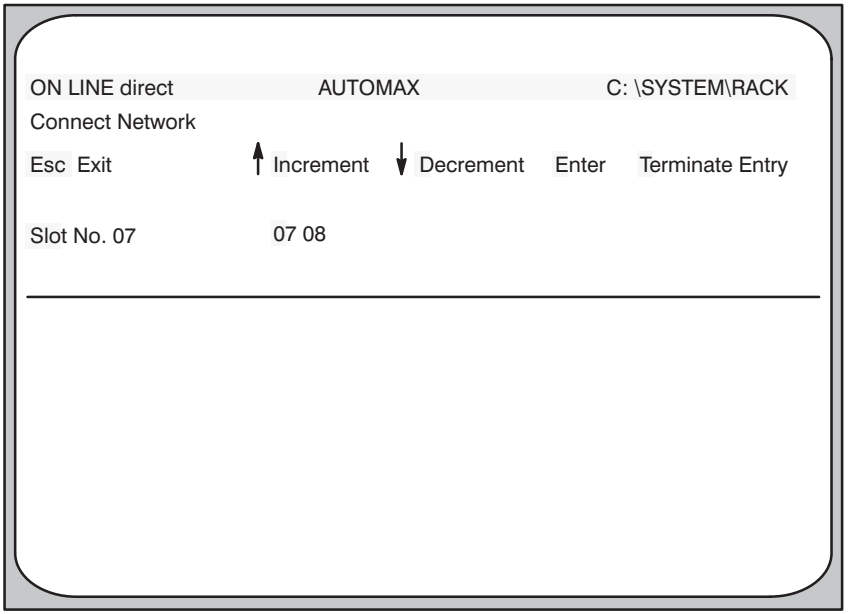


Figure 12.2 - ON LINE CONNECT NETWORK Menu (Select Network)

If you are using serial communication, you must select the slot in the local rack that contains the Network Communications module (M/N 57C404A and later) which is on the network that contains the rack you want to communicate with. Use the up and down arrows to scroll through the available slots, which are displayed on the screen, or type the number of the slot. When you have selected a slot, enter <CR>. The Executive software will display the screen shown in figure 12.3.

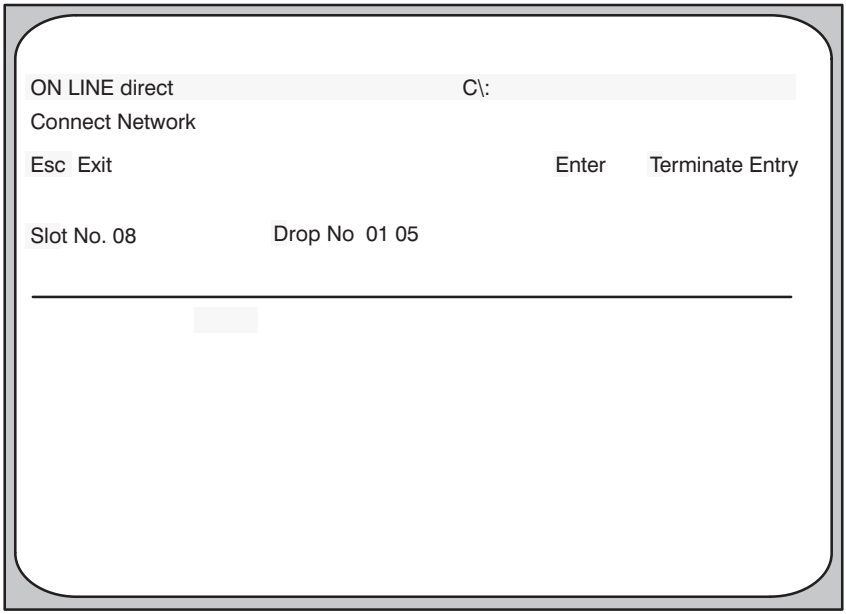


Figure 12.3 - ON LINE CONNECT NETWORK Menu (Select Network Rack)

From the screen shown in figure 12.3, select the rack (drop) number on the network. The drop numbers available are shown in blocks of 15. You can use the PgUp and PgDn keys to see other blocks of drops, if applicable. Enter the drop number you wish to connect to and <CR>. From this point until you change your connection through the ON LINE CONNECT menu again, all communications will take place with the network drop chosen. The top line of most ON LINE screens will display the drop number of the rack with which the personal computer is communicating.

12.6 Displaying the Network Connection Table

Up to four users can be connected to the same AutoMax rack over the network. Typing “U” for “Userlist” from the Connect menu shown in figure 12.1 allows you to display the Network User List. The Network User List displays users who are connected to the network. See figure 12.4.

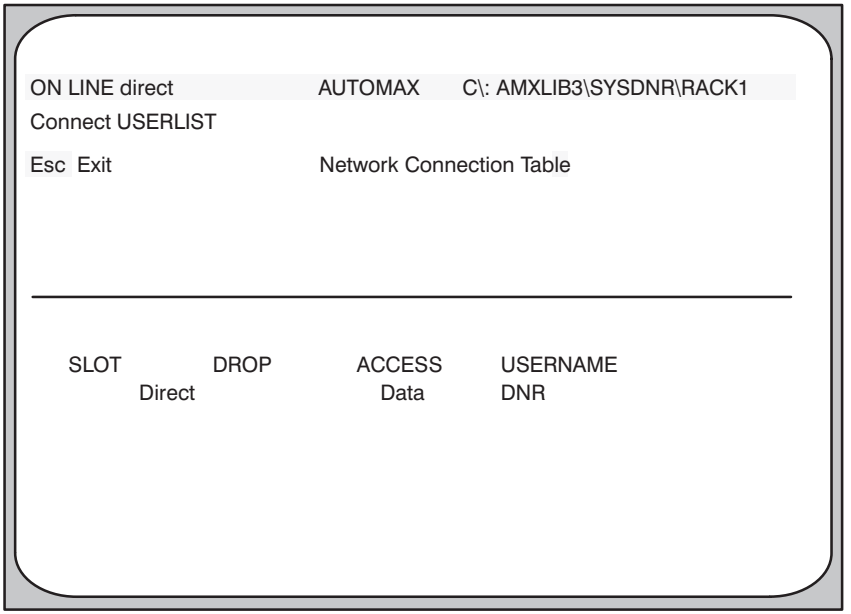


Figure 12.4 - Network Connection Table

The SLOT column shows the number of the slot that contains the Network module that is being used to communicate with the rack (drop). If the user is using direct communication with the local rack, this field will display DIRECT.

The DROP column shows the number of the network drop to which the user is connected. If the user is using direct communication with the local rack, this field will display DIRECT.

The ACCESS column shows the user's access level (None, Data, Task, or Rack). See section 9.2 for a description of access levels.

The USERNAME column shows the name that was entered in AutoMax Setup.

13.0 ON LINE MENU: INFO/LOG

Entering “I” for Info/Log from the ON LINE menu allows you to display information about the system software on any AutoMax Processor or UDC module in the rack or to view the status and error log, if any, for a selected task.

13.1 Info/Log Processor and UDC Information Display

If you select Info/Log without having selected a task by using the Enter key or <CR>, AutoMax will assume you want to display information about an AutoMax Processor and the Common Memory module (M/N 57C413B or 57C423) or a UDC module. You will be prompted for the slot number of the AutoMax Processor or UDC module of interest.

AutoMax Processor

Figure 13.1 shows a sample display for an AutoMax Processor in slot 1 with no Common Memory module.

```
ON LINE direct                    AUTOMAX      C: \LIB\SYS\RACK
Info/Log

Esc  Exit          Control Processor Information for Slot 1


                   Software ID:          422526-001C STD

                   Clock Tick (micro sec):     5500

                   CPU Utilization:            34%

                        Memory      Memory       Largest
                         Used        Free      Free Block    Battery
PROCESSOR               2924        151280      151280       Replace
COMMON                   N/A          N/A         N/A          N/A
```

Figure 13.1 - Info/Log Processor Information Display

The “Software ID” depends on the version of the software (operating system) that has been loaded to the AutoMax Processor. (Refer to section 1.4.2 for a list of the software part numbers for versions of the AutoMax Programming Executive.) If the Ethernet version of the operating system is loaded, “ENET” will be displayed on the line as well. “STD” will be displayed for the standard operating system. “Clock Tick” refers to the speed of the real-time clock. “CPU Utilization” shows the percentage of the AutoMax Processor currently being used to run application tasks. Note that the CPU Utilization percentage is measured over a two second time window. This value should be kept under 80% to ensure that there are no overlaps in AutoMax task execution.

Memory statistics for the AutoMax Processor and Common Memory module are given in bytes. The field “Largest Free block” refers to the largest continuous block of memory available. The “Battery” field refers to the status of the on-board battery.

UDC Module

Figure 13.2 shows a sample display for a UDC module in slot 6.

```
ON LINE direct                    AUTOMAX      C: \RPE\V33
Info/Log

Esc  Exit          UDC Information for Slot 4

_____________________________________________________________

              Software ID:    422544-044A Tick Rate (micro sec): 500

              Parameter File Date/Time: 03-08-1993 10:59:48a

              UDC Utilization: 20%

                    Memory      Memory
                     Used        Free
Task A              2642        17838       Bytes
Task B              0           20480       Bytes
COMMON              128         12160       Bytes
PMI-A OS Status: SD3000                     PMI-B OS Status: SV3000
PMI-A Parameters: Installed                 PMI-B Parameters: Installed
PMI-A Slot 0: PMI Processor                 PMI-B Slot 0: PMI Processor
PMI-A Slot 1: Resolver & Drive I/O          PMI-B Slot 1: Resolver & Drive I/O
PMI-A Slot 2: DC Power Technology           PMI-B Slot SV3000
```

Figure 13.2 - Info/Log UDC Information Display

The “Software ID” depends on the software (operating system) and hardware version of the UDC module. “Tick Rate” refers to the speed of the real-time clock in the UDC module. The “Parameter File Date/Time” will display the most recent date and time that the parameter object file was generated. “UDC Utilization” shows the percentage of the UDC CPU’s resources being used to run UDC tasks. The UDC Utilization percentage is measured over a two second time window. This value should be kept under 75% to ensure that there are no overlaps in UDC task execution. Memory statistics displayed for the UDC module are given in bytes.

The “PMI OS Status” fields show whether the PMI operating system has been loaded to each of the PMI Processors (Not Loaded or the OS type (e.g., SA3000)). If the PMI Processor is not connected to the UDC module, this field will display “Not Connected”. The “PMI Parameters” fields show whether the drive parameters have been loaded to each of the PMI Processors (Installed or Not Installed). If the PMI OS Status field is displaying “Not Connected”, the PMI Parameters fields will be blank. If the PMI Processor is connected to the UDC module, the modules contained in the PMI rack(s) will be listed.

13.2 Info/Log Task Information Display

If you select a task using the Enter or <CR> before selecting Info/Log, AutoMax will display the task status and the error log, if any, for the AutoMax or UDC task. A sample display is shown in figure 13.3.

```
ON LINE direct                    AUTOMAX      C: \LIB\SYS\RACK
Info/Log
Esc  Exit

F1  Help

SLOT      NAME                TYPE     PRI    STATUS    SOURCE DATE
  5       APP1B               Blk      B      Stop      08-31-1993  02:08:18p

ERROR LOG

      NUMBER       STATEMENT      DATE       TIME      ERROR SPECIFIC
   49  =  31 HEX       0        No date     00:00       0 =     0 HEX
  774 = 306 HEX       0        No date     00:00       0 =     0 HEX
            Precision lost in integer control variable to real conversion
  775 = 307 HEX       0        No date     00:00       0 =     0 HEX
            Real to integer overflow in FOR statement TO value


 Error 31:    Slot=0    Bit=0   (Reg based on type of card in slot)
 Net: drop=0,    reg=0          Remote I/O:drop=0,    slot=0,     reg=0
 Modbus: reg=n/a AutoMate: reg=n/a A-B: file= n/a, reg=n/a Local I/O: reg=0
```

Figure 13.3 - Info/Log Task Information Display

The status line for the task displays the slot of the Processor or UDC module on which the task is loaded in the SLOT field. The task name is shown in the NAME field. The task type (.BAS, .BLK, or .PC) is shown in the TYPE field. The priority of AutoMax Processor tasks, assigned when the task is added to the rack, is shown in the PRI field. For UDC tasks, the drive (A or B) the task is assigned to will be shown. The status of the task, either Running, Stopped or in Error, is shown in the STATUS field. The date that the source file was last edited is shown in the SOURCE DATE field. In the sample display, the .PC task FRCS1LAD loaded on the Processor in slot 1 is designated as priority 7 and is currently running.

If any errors occurred while the task was running, the STATUS field will display Error. An error log is generated and shown on the screen along with the task information. A maximum of three error log entries are saved, in the order of occurrence, for each task. Only the first, second, and last errors that occur will be stored. Any other errors will not be stored.

For each error, the error log displays the statement number where the error occurred (STATEMENT) if it is relevant and can be determined. The DATE and TIME fields will be blank; these functions are not currently supported. There may be other information supplied for certain errors in the ERROR SPECIFIC field. “31” errors are decoded at the bottom of the Error Log. Note that the error log for AutoMax tasks is maintained through a power cycle; the error log for UDC Control Block tasks is not maintained through a power cycle. The UDC error log is cleared when power is removed from the rack. For more information about clearing the error log, see 19.0.

Like AutoMax tasks, UDC tasks can also access the error log by using the BASIC function TST_ERRLOG@ and the statement CLR_ERRLOG@.

14.0 ON LINE MENU: TRANSFER

Entering “T” for “Transfer” from the ON LINE menu allows you to change the default path, view the contents of the default path, load application tasks onto the rack, and save application tasks from the rack. Sections 14.1-14.4 describe these options in more detail. See figure 14.1 for the Transfer menu.

Figure 14.1 - TRANSFER Menu

14.1 Changing the Default Path

You can change the default DISK:\LIBRARY\SYSTEM\RACK, or path, by selecting “P” from the TRANSFER Menu. The default path is shown in the upper right hand corner of the screen. The current default determines the location to which you save tasks from an AutoMax Processor or UDC module and from which you load tasks to a Processor or UDC module.

Note carefully that your path must be the path in which the configuration and application task files for the rack to which you are connected are stored. If the path is incorrect, you may load the wrong configuration and application task files to the rack. What’s more, if you save back application tasks and configuration files, you will write over the configuration object file in the current path. You may also write over application tasks that have the same name. If you copy over the configuration file, you can generate it again because the configuration database is not overwritten.

14.2 Directory of the Default Path

Selecting “F” for “Files” from the TRANSFER menu displays the directory of the current default path. To change the default, see section 14.1.

14.3 Loading the Rack Configuration, Drive Parameters, and Tasks onto the Rack

The Load option from the Transfer menu allows you to load the rack configuration, drive parameters, and application tasks onto all AutoMax Processor and UDC modules in the rack and all Processor modules on the network(s) through the connection at the leftmost Processor module. BASIC and Control Block tasks must be compiled first. Drive parameter and configuration files must be generated first. Note that a maximum of 32 tasks can be loaded to an AutoMax rack. The keyswitch must be in the PROGRAM position and the password must be entered to load any tasks. All tasks will be loaded onto the Processor specified when the task was added to the rack in the Task Manager application. Utility tasks must be loaded separately even though they are added to the rack like other tasks. See 14.3.1 for more information.

After tasks are loaded onto the rack, they are installed by the operating system. This procedure involves verifying that all physical I/O points in the system are defined and consistent with the information in the configuration.

When you select “L” for “Load” from the Transfer menu, the screen will display the following choices:

- Normal Rack Configuration
- Debug Rack Configuration
- Every Parameter File
- Single Parameter File
- Tasks
- All

You have two options when loading the **rack configuration**. You can load the complete configuration (normal) or you can load a test configuration with all I/O mapped to memory (debug). The rack configuration must be loaded before any AutoMax tasks can be loaded to the rack.

If you want to use the test configuration with AutoMax and UDC tasks, you must select the “All” option (described below) to load the drive parameters, AutoMax tasks, and UDC tasks. Then return to the Transfer menu and select Load “Debug Rack Configuration” to load the test configuration to the rack.

You also have two options when loading the **drive parameters** to the UDC module(s). You can load the drive parameters to a UDC module in a specific slot in the rack, or to all UDC modules in the rack. The drive parameters must be loaded before the UDC tasks are loaded. Note that drive parameters cannot be loaded to a UDC module if UDC tasks are running on the target UDC module. You must stop the UDC tasks for the target UDC module before you can load the drive parameters.

Drive parameters are maintained through a power cycle. If changes to the PMI D/A port parameters are made online, these parameters will revert to those saved in memory after power is cycled. To make changes to the drive parameters permanent, you must change the parameters using the Rack Configurator and then re-load them. See the appropriate DPS Configuration and Programming instruction manual for more information.

If you select **Tasks**, the screen will display a list of all the AutoMax and UDC application tasks for the rack. The task files listed on the screen are either AutoMax Ladder Logic tasks, which do not need to be compiled, or the object code files that were created when you compiled AutoMax BASIC and Control Block tasks or UDC Control Block tasks. Select the task you want to load from the list. A log file will automatically be created that will list any errors that occurred while the task was being loaded. The log file will have the same name as the task, and the extension .LOG. Remember that you must load the rack configuration before you can load any AutoMax tasks to the rack. You must load the rack configuration and the drive parameters before loading UDC tasks to the UDC module.

It is permissible to load tasks onto a rack that is currently running tasks. If the task is new, i.e., has a unique name, you can load the task without stopping tasks already running. AutoMax Executive V3.4 will prompt for the slot and priority of the task. Note that a CRITICAL task cannot be loaded until all tasks currently in RUN are stopped.

If you wish to load a task with the same name as one currently running in the rack, you must stop the task before you will be permitted to load the new task. Note that any task you load will not go into RUN unless you put it into RUN.

If you wish to load a UDC task for a drive that does not have a task running, you can load the task without stopping the one curently running on the UDC module.

If you wish to load a new UDC task for a drive that already has a task running, you must stop the one currently running on the UDC module before you will be permitted to load the new UDC task.

If you select **All**, the normal rack configuration, the drive parameters for all the UDCs in the rack, all the AutoMax application tasks (except utility tasks), and all UDC tasks will be loaded automatically. The current rack configuration and drive parameters (if any) will be written over. Any application tasks currently on the AutoMax Processors or UDC modules whose names are the same as the ones being loaded will be written over.

.LOG and .MEM Options

If you select normal rack configuration, debug rack configuration, or all, you will have the option to also generate .LOG and .MEM files. The .LOG option creates a file that contains a log of any errors that occurred while tasks were being loaded. The .LOG file will have the same name as the configuration file, with the extension .LOG.

If you select the .MEM option, you are disabling the utility that loads tasks at higher speed. This option frees up six (6) bytes of Processor memory per common symbol used.

14.3.1 Utility Tasks

Tasks designated as utility tasks when added to the rack are usually used for testing purposes and not for application control. Utility tasks can only be loaded into AutoMax Processor modules. They cannot be loaded onto UDC modules.

The object code file for such tasks can be loaded onto the rack just like the object code for other tasks. Utility tasks must be loaded individually.

To load a utility task, enter the name of the task at the filename prompt. You will then be prompted for the slot number of the Processor on which the task is to be loaded (0-4), and the task priority (4-11). Note that this information was already entered when the task was added. It is required here as well.

14.4 Saving Tasks from the Rack

You can save tasks from any AutoMax Processor or UDC module in the network to the default path through the single connection at the leftmost Processor. You must obtain task access for the tasks you want to save from the rack.

Entering “S” for Save at the Transfer menu brings up a list of tasks that may be saved from the rack. This list contains all tasks in the rack, whether they are running or stopped, including utility tasks. Tasks can be saved either individually by entering the task name, or all (all tasks listed on the screen) at one time by entering the name of the rack configuration file and the /All option described in section 14.4.1.

You have three other options when saving tasks from the rack: /Tunable, /Include, and /Log. These options, as well as /All, are described below in more detail.

Note that you can only save tasks from the rack if you elected to make them reconstructible when you compiled the task originally. Ladder Logic tasks are always reconstructible. Note carefully that your path must be the path in which the configuration and application task files for the rack to which you are connected are stored. Otherwise, it is possible to write over files in the current path. If your path is incorrect, any task you save back from the Processor cannot be used until you add it to the rack using the Task Manager.

14.4.1 Save Option: /All

You can save all the reconstructible tasks in the rack, including utility tasks, using the /All option. Enter the name of the configuration task (_CONF.CNF) at the filename prompt and then /All at the option prompt.

14.4.2 Save Option: /Tunable

You can save only the tunable variable values in a particular task using the /Tunable option. Enter the name of the task at the filename prompt and /Tunable at the option prompt. The list of values will be saved to the default path with the same filename as the task, but with the extension .TUN. If you specify the /All option as well as the /Tunable option, the tunable values will be written to separate .TUN files. You can view or edit a .TUN file using the text editor. The task does not need to be reconstructible to save tunable variables.

14.4.3 Save Option: /Include

You can save only the tunable variable values in a particular task using the /Include option. Enter the name of the task at the filename prompt and /Include at the option prompt. The list of values will be saved to the default path with the same filename as the task, but with the extension .INC. If you specify the /All option as well as the /Include option, the tunable values will be written to separate .INC files. You can view or edit a .INC file using the text editor. The task does not need to be reconstructible to save tunable values using this option. Note, however, that the .INC file being created will contain only the tunable variable values. If a .INC file with the same name already exists on the personal computer, it will be written over. If the original .INC file contained information other than tunable variable values, this information will be lost.

14.4.4 Save Option: /Log

The /Log option is used to generate a log file of any errors that occurred while tasks were being saved from the AutoMax Processor or UDC module. If you enter the configuration file name at the filename prompt and specify the /All option as well as the /Log option, the log file will have the same name as the configuration file, but with the extension .LOG. If you entered a specific filename at the prompt, then the .LOG file will have the same name as that file, and the extension .LOG. You can view or edit a .LOG file using the text editor.

14.5 Recovering a Rack

The procedure that follows describes how to recover the configuration information and tasks for a rack from the AutoMax Processor(s). It is possible that the configuration information and tasks on the AutoMax Processor(s) could be different than the information on your personal computer. This situation could occur if the rack configuration was changed using another programming terminal, and then the changes were not copied to the original system, section, and rack. In this case, the latest rack configuration and task information would exist only on the AutoMax Processors in the rack. This should not normally occur.

Note that the rack to be recovered will be treated the same as an AutoMax V2 rack that is being imported. This means that you will lose the module data, but recover the variable mapping. Before beginning, read sections 5.3.9 through 5.3.9.6. These sections describe how to import an AutoMax V2 rack, which is the second part of the rack recovery procedure. Note that this procedure will recover

the rack configuration and the tasks only if they were designated as reconstructible. If the rack configuration was not designated as reconstructible, it cannot be recovered. Also, only reconstructible tasks will be saved back to the personal computer.

Step 1. Start at the DOS prompt in the root directory. Use the DOS MAKE DIRECTORY command to create a system and rack subdirectory, e.g., <DRIVE>:<SYSTEM_NAME>\<RACK_NAME>.

Step 2. Access the AutoMax Programming Executive.

Step 3. Go online and then access the Transfer menu.

Step 4. Select Path from the Transfer menu. Change the path to the system\rack subdirectory you created in step 1.

Step 5. Select Save from the Transfer menu. Use the Save All option described in 14.4.1 to save the rack configuration and all the reconstructible tasks to the system\rack subdirectory you created in step 1.

Step 6. Return to the System Configurator. Use the Remove command to delete the AutoMax rack you are recovering. You will re-create the rack in the next step.

Step 7. Use the Import Rack procedure described in section 5.3.9.6 to re-create the rack. When you are prompted for the source directory path, enter the system\rack path you created in step 1.

15.0 ON LINE MENU: RUNNING TASKS

WARNING

UNDERSTAND THE APPLICATION BEFORE STARTING A TASK. OUTPUTS MAY CHANGE STATE, RESULTING IN MACHINE MOVEMENT. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

IT IS THE RESPONSIBILITY OF THE USER TO ENSURE SAFE OPERATION OF THE APPLICATION PROCESS SHOULD THE APPLICATION TASKS BE STARTED OUT OF SEQUENCE. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

THE USER MUST PROVIDE AN EXTERNAL, HARDWIRED EMERGENCY STOP CIRCUIT OUTSIDE THE CONTROLLER CIRCUITRY. THIS CIRCUIT MUST DISABLE THE SYSTEM IN CASE OF IMPROPER OPERATION. UNCONTROLLED MACHINE OPERATION MAY RESULT IF THIS PROCEDURE IS NOT FOLLOWED. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.

Note that a task may need to perform initialization before entering the main loop or the scanned portion of the task. When the task is started, this initialization will pre-empt any task which may be running at a lower priority. Depending on the application, this may result in a STOP-ALL error.

The RUN option from the ON LINE menu is used to put AutoMax and UDC tasks loaded onto the rack into run. You can put tasks into run individually, or you can put all tasks in the rack into run. Utility tasks will also be put into run by running all tasks as described in 15.2. The keyswitch must be in the PROGRAM position and the password must be entered to run any task. You must also obtain either rack access or task access for all the tasks you want to run.

Your first step in running tasks depends on whether you want to run one task only or all the tasks in the rack. Follow the directions in 15.1 for running one task only. Follow the directions in 15.2 for running all tasks specified in the configuration for the rack.

15.1 Running an Individual Task

WARNING

THIS COMMAND OVERRIDES THE NORMAL STARTING SEQUENCE OF THE APPLICATION TASKS, WHICH IS DETERMINED BY THE PRIORITY ASSIGNED WHEN THEY ARE ADDED TO THE RACK. THE RUN ALL OPTION SHOULD NORMALLY BE USED TO START TASKS. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.

Follow the directions below to put an individual task into run:

1. Select the task to be run from those displayed on the ON LINE menu. To select the task, use the up and down arrow keys to move the “>” pointer to the desired task.
2. Type the “Enter” or <CR> key. The selected task will be shown in reverse video.
3. Type “R” to put the task into run. The task “Status” field on the display will show that the task is in RUN.

Note that UDC tasks can be run regardless of whether the PMI is communicating with the UDC.

15.2 Running All Tasks in the Rack

To run all AutoMax and UDC tasks in the rack, including utility tasks, using the procedure described in this section, all tasks in the rack except utility tasks must be loaded in the rack. If this is not the case, the screen will display an error message and no tasks will be put into run.

Follow the directions below to put all tasks in the rack into run:

1. Do not select an individual task while at the ON LINE menu. Instead, type “R” from the ON LINE menu with no task selected. This results in the display shown in figure 15.1.

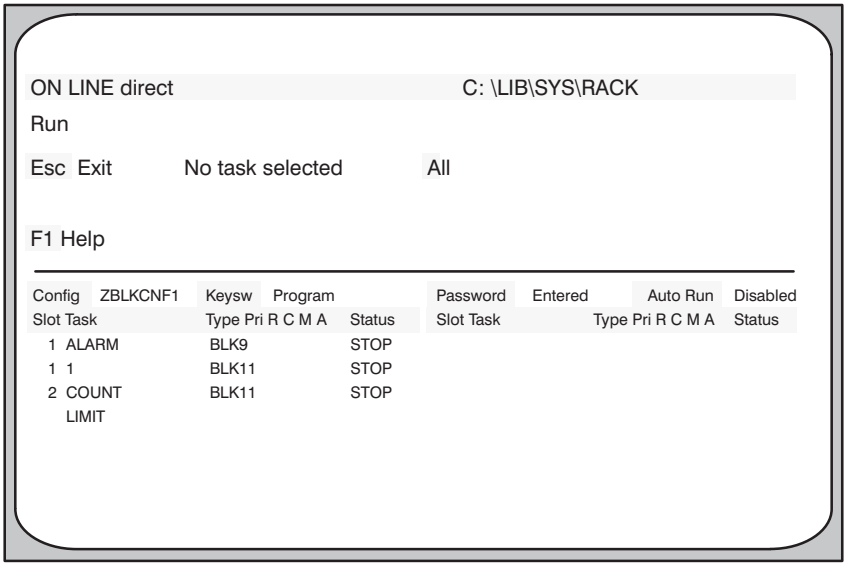


Figure 15.1 - RUN ALL Display

2. To run all the tasks, type “A” for all. The “Status” field will show all tasks in RUN. AutoMax Tasks are started in order of priority from the highest priority (4) to the lowest priority (11). Tasks at the same priority level are started in alphabetical order. Tasks on different Processors are started independently of each other. UDC tasks are always run drive A task first, followed by the drive B task.

16.0 ON LINE MENU: STOPPING TASKS

WARNING

IT IS THE RESPONSIBILITY OF THE USER TO ENSURE THAT THE APPLICATION PROCESS STOPS IN A SAFE MANNER WHEN THE APPLICATION PROGRAMS STOP. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

You can stop one or all tasks running in the rack using the stop command from the ON LINE menu. To stop any tasks, the keyswitch must be in the PROGRAM position and the password must have been entered.

Note that you cannot stop individual tasks that were specified as CRITICAL when added to the rack in the Task Manager. Critical task status display described in 11.3. If you want to stop a task that is critical, you must stop all tasks in the rack. See 16.1 for stopping an individual task and 16.2 for stopping all tasks in the rack.

16.1 Stopping an Individual Task

WARNING

DEPENDING ON THE APPLICATION, STOPPING AN INDIVIDUAL TASK MAY RESULT IN LOSS OF CONTROL OF THE APPLICATION PROCESS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. IT IS RECOMMENDED THAT THE STOP-ALL COMMAND BE USED TO STOP TASKS NORMALLY. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

You can stop any individual task that is not designated with a “C” for CRITICAL in the C field of the task status display. Note that outputs controlled by the selected task will retain their last state. To stop an individual task, follow the directions below.

1. Select the task from the ON LINE menu display using the arrow keys to move the “>” pointer.
2. Type “Enter” or <CR> to select the task. The selected task will be shown in reverse video.
3. Type “S” to stop the task. The task “Status” field on the display will show that the task is in STOP.

16.2 Stopping All Tasks in the Rack

When you stop all tasks in the rack by issuing a STOP-ALL command (or the system issues a STOP-ALL due to an error in the rack), all tasks that are running will be stopped, including utility tasks (see 14.3.1). The system will issue a clear command to all Reliance I/O modules in the local and remote chassis, including I/O connected to the PMI rail ports. See table 16.1 for a summary of what happens

when a STOP-ALL command is issued by the operator or the system issues a STOP-ALL. To stop all tasks in the rack, follow the directions below:

1. From the ON LINE menu, select “S” for Stop.
2. The screen will display “No task selected” and the option “All”. Enter “A” to stop all tasks. The “Status” field for all tasks will then show that the tasks are in STOP.

Table 16.1- Status of Data in the AutoMax Rack after a STOP-ALL Command or STOP-ALL Fault

	AutoMax Processor	**UDC Module**	**PMI Processor**
LOCAL tunable variables	retained	retained	retained
LOCAL variables	retained	reset to 0	N/A
COMMON memory variables	non-volatile are retained; others are reset to 0	N/A	N/A
I/O variables (including UDC dual port memory)	inputs retained and updated; outputs are reset to 0	inputs retained and updated; outputs are reset to 0	all I/O is reset to 0
Input values, including: Feedback registers UDC/PMI communication status registers UDC Error Log info	retained	retained	N/A
Output values, including: Command registers Application registers ISCR registers Scan-per-interrupt register Scans-per-interrupt counter	reset to 0	reset to 0	N/A
Parameter configuration variables	N/A	retained	N/A
UDC test switch information	N/A	retained	N/A
D/A setup configuration	N/A	retained	N/A
Operating system	retained	retained	retained

17.0 ON LINE MENU: DELETING TASKS

You can delete one or all tasks in the rack using Delete function from the ON LINE menu. Note that a task must be stopped before it can be deleted. The keyswitch must be in the PROGRAM position and the password must have been entered. You must have either rack access or task access to all of the tasks you want to delete. See 17.1 for deleting one task from the rack and 17.2 for deleting all tasks from the rack.

17.1 Deleting an Individual Task

To delete one task from the rack, follow the directions below.

1. Use the arrow keys to move the “>” pointer to the desired task.
2. Type “Enter” or <CR> to select the task. The selected task will be shown in reverse video.
3. Delete the task by entering “D” for Delete. The system will prompt you with “are you sure?” before deleting any tasks. Answer “Y” for yes and “N” for no. The deleted task will be erased from the AutoMax Processor or UDC module and will no longer appear on the display.

17.2 Deleting All Tasks from the Rack

To delete all tasks in the rack follow the directions below.

1. Enter “D” for Delete without selecting any tasks first.
2. Enter “A” to delete all tasks in the rack. The system will prompt you with “are you sure?” before deleting any tasks. Answer “Y” for yes and “N” for no. All tasks will be erased from the AutoMax Processors and UDC modules and will disappear from the display.

18.0 ON LINE MENU: MONITORING AND EDITING TASKS

AutoMax allows you to monitor and change the status of variables and I/O points, display and modify Ladder Logic sequences in real time, and force and unforce variables. In order to use any options except monitoring and adjusting tunable variables, the keyswitch must be in PROGRAM and the password must have been entered.

To begin monitoring, enter “M” for Monitor from the ON LINE menu. The resulting **Monitor** menu shown in figure 18.1 in turn allows you to access six other menus which are described below.

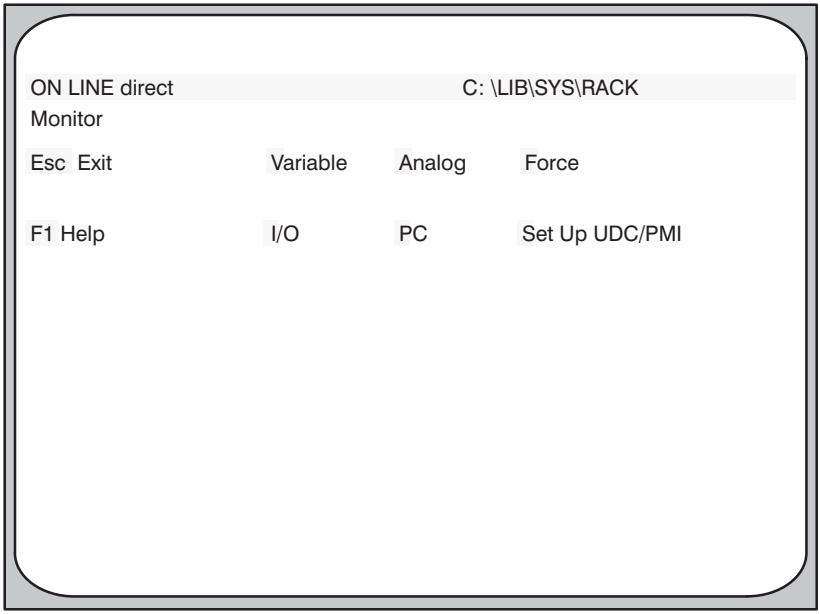


Figure 18.1 - Monitor Menu

Monitor Variable: used to display continuously current variable values and to change, force, and unforce variables. See 18.1 for more information.

Monitor I/O: used to output continuously the state of I/O registers and make changes to those registers. See 18.2 for more information.

Monitor Analog: used to output continuously up to two integer variables through the D/A converters located on a Drive Analog I/O module (B/M 57405). See 18.3 for more information.

Monitor PC: used to display continuously the state of each element in a sequence of a Ladder Logic task and do online editing of the task being monitored. See 18.4 for more information.

Monitor Force: used to force variables to a specific value. Once a variable is forced, it will retain the forced value until it is unforced. See 18.5 for more information.

Monitor Set Up UDC/PMI: used to select variables to drive the D/A analog outputs on the UDC and PMI Processors and to enter minimum and maximum values for scaling. See 18.6 for more information.

The six menus are identified in the upper left hand corner of the screen with the word "**Monitor**" followed by the kind of monitor, e.g. "**Variable**". Depending upon the option you choose from the menu, the field may display even more detail, e.g., "**Monitor Variable Modify**". The names of sub-menus available from the Monitor menu will be shown in boldface in this instruction manual to avoid any confusion between menu titles and the functions permitted from those menus.

The steps which describe the operations you can perform in the various **Monitor** menus are given in the order in which AutoMax prompts you for the information needed, such as variable name, register number, etc. In general, you will be prompted to enter the information from left to right into fields displayed on the screen in reverse video with a blinking cursor immediately following.

When you are finished entering the information for a particular field, type "Enter" or <CR> to indicate you are finished and the cursor will immediately move to the next field. You can use the arrow keys to move the cursor within the fields. You can also use the cursor to edit your entry if you have not completed entering all the required information for the option.

At times, certain fields will show a default. This default is usually the information you entered the last time you used the particular option. To select the default, type "Enter" or <CR> and go on to the next field. If you do not wish to select the default, simply type over the default.

In general, when you are finished entering the information required for the option, you must type "Enter" or <CR> to execute that option.

18.1 Monitoring and Modifying Variables

The **Monitor Variable** menu allows each user (up to four per rack) to monitor up to 16 COMMON and/or LOCAL variables in any combination of boolean, single precision integer, double precision integer, and real. For each UDC module in the rack, up to 32 LOCAL variables can be monitored, regardless of the number of users (16 maximum per user).

Entering "V" for Variable from the **Monitor** menu displays the **Monitor Variable** menu, shown in figure 18.2.

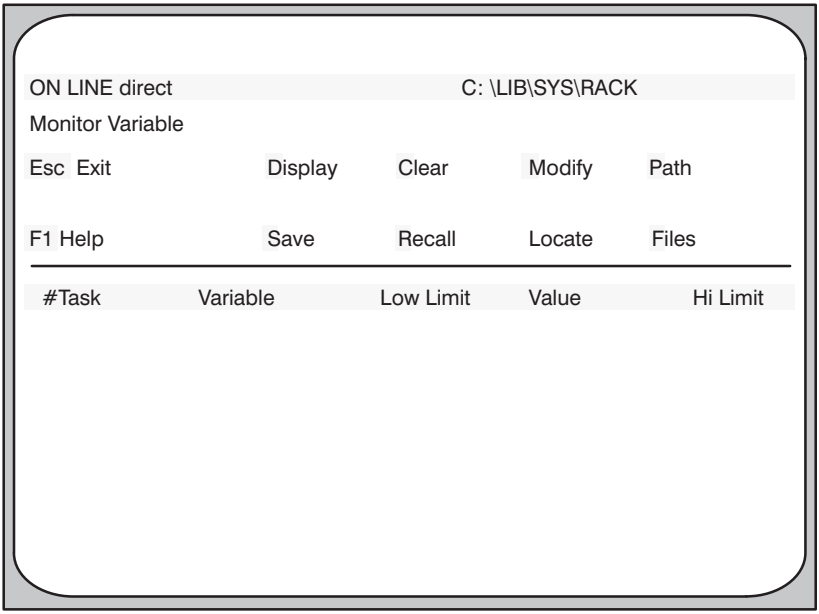


Figure 18.2 - Monitor Variable Sub-Menu

The lower portion of the **Monitor Variable** screen display, which is updated continually, includes the following fields:

(#)	- Display number (screen position).
Task	- Identifies the name of the task in which the variable is defined. This field is blank if the variable is COMMON.
Variable	- Variable name, including type character.
Low Limit	- Low limit for tunable variables. This field is blank for all other variables.
Value	- Current value of the variable. The value is updated in real time.
Hi Limit	- High limit for tunable variables. This field is blank for all other variables.

The **Monitor Variable** menu allows you to choose from the following eight options:

Display: Adds a variable to the display list. See 18.1.1 for more information.

Clear: Removes a variable or all variables from the display list. See 18.1.2 for more information.

Modify: Allows you to modify a variable. See 18.1.6 for more information.

Save: Allows you to save a display list to your default path. See 18.1.3 for more information.

Recall: Allows you to recall a display list from your default path. See 18.1.4 for more information.

Locate: Allows you to display the variable name associated with an I/O address or display the I/O address associated with a variable name. See 18.1.5 for more information.

Path: Allows you to change the path.

Files: Allows you to display a directory of the files in the current path.

18.1.1 Displaying a Variable

Follow the directions below to display a variable on the screen.

1. Enter “D” for “Display” from the Monitor Variable menu.
2. Enter the name of the task in which the variable in question is defined, followed by “Enter” or <CR>. If the variable is COMMON, use the “Enter” or <CR> key to skip over the field.
3. Enter the name of the variable, complete with any terminating character designating variable type (“@” for boolean, “%” for single precision integers, or “!” for double precision integers).
4. Enter the desired format of the display, choosing from the options listed below. The variable will be shown in the display.

“D” for Decimal
“B” for Binary
“H” for Hexadecimal
— see figure 18.3 for more information

Variables can be displayed in the following formats:

Variable Type	Format	Range of Values
Boolean	Boolean	TRUE/FALSE
Single Integer	Decimal	-32767 to +32767
	Hexadecimal	0 to FFFF
	Binary	0000000000000000 to 1111111111111111
Double Integer	Decimal	-2147483648 to +2147483647
Real	Decimal	+5.42101070E-20 to +9.22337177E18
		-2.71050535E-20 to -9.22337177E18

Figure 18.3 - Variable Display Format

Note that if the display is full (16 variables) and you attempt to add another variable to the list, you will be prompted to delete a variable from the list. The selected variable will be deleted, and the new variable will be added at the end of the list.

18.1.2 Clearing a Variable from the Display

Follow the directions below to clear a variable from the display.

1. Enter “C” for Clear from the **Monitor Variable** menu.
2. To clear one variable from the screen, enter the display position number (1-16) of the variable at the blinking cursor. To clear all variables from the screen, enter “A” for All.
3. Type “Enter” or <CR>. The variable will be cleared from the display.

18.1.3 Saving a Monitor Variable Display List

You can save the variables listed on the **Monitor Variable** display to the default path displayed in the upper right hand corner of the screen. Only the list of variables is saved, not the variable values. If your default destination is a floppy disk, it must not be write-protected.

You can give each list a standard 8-character filename. The AutoMax Executive software automatically attaches the file extension $MV. You can later recall the list to the screen using the Recall option described in 18.1.5.

To save the current **Monitor Variable** list, follow the directions below:

1. If you have not already done so, enter “S” for Save from the **Monitor Variable** menu shown in figure 18.2.
2. Enter the filename (up to 8 characters) for the display list. Do not attach a file extension. The file will automatically be given the extension $MV.
3. Type “Enter” or <CR>.
4. You can recall the screen display using the Recall option described in 18.1.4.

18.1.4 Recalling a Monitor Variable Display List

You can recall any previously saved **Monitor Variable** display list. To recall the **Monitor Variable** display list, follow the directions below.

1. From the Monitor Variable menu, select “R” for Recall.
2. Enter the name of the display list to recall,
3. Type “Enter” or <CR>.

18.1.5 Locating a Variable or I/O Address

The Locate command can be used to display the variable name associated with an I/O address and display the I/O address associated with a variable name. When you are locating a variable, you have the option of finding the variable using a local I/O address (slot, register) or a remote I/O address (slot, drop, slave slot, register). The information will be retrieved from the configuration file stored in the rack. Note that if you are using interface modules to communicate with foreign I/O, you will not be able to use this command to display Modbus, AutoMate, or A-B register numbers. This command **will** display Modbus, AutoMate, and A-B registers using the equivalent **Multibus** addresses.

Step 1. Select “L” for Locate from the Monitor Variable menu.

Step 2. Select “V” for Variable to locate the variable name. Continue with step 3.

or

Select “I” for I/O to locate the variable’s I/O address. Skip to step 5.

Step 3. Select either “L” for Local I/O Address or “R” for Remote I/O Address.

Step 4. If you selected Local I/O Address, enter the slot, register, and bit number (for boolean variables). If you selected Remote I/O Address, enter the slot, drop, slave slot, register, and bit number (for boolean variables).

The name of the variable that is assigned to the address entered will be displayed.

If there are no variables mapped to the specified I/O point, then the message “There is no variable mapped to this I/O” will be displayed.

Step 5. Enter the variable name.

If the variable is a boolean mapped to an I/O point, the local I/O address (slot, register, bit) and remote I/O address (slot, drop, slave slot, register, bit) will be displayed.

If the variable is an integer, double, or real mapped to an I/O point, the local I/O address (slot, register) and remote I/O address (slot, drop, slave slot, register) will be displayed.

If the variable is not in the rack configuration, the message “Variable is either not found or is a LOCAL variable” will be displayed.

If the variable is mapped to a memory location, the message “Variable is a common memory variable. No I/O address is available” will be displayed.

18.1.6 Modifying a Variable

WARNING

DEPENDING ON THE APPLICATION, THE USE OF THE TUNE FUNCTION MAY RESULT IN INSTABILITY OF THE APPLICATION PROCESS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

THE SET AND FORCE FUNCTIONS BYPASS CONTROL OF THE APPLICATION PROCESS BY THE APPLICATION TASKS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

VARIABLES AND OUTPUTS WHICH ARE FORCED BEFORE AC POWER IS LOST WILL REMAIN FORCED WHEN AC POWER IS RESTORED. SHOULD AC POWER BE LOST WHILE VARIABLES ARE FORCED, THE USER MUST ENSURE THAT UNEXPECTED MACHINE MOVEMENT DOES NOT OCCUR WHEN AC POWER IS RESTORED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

Selecting “M” for Modify from the **Monitor Variable** menu allows you to set, tune, force, or unforce a variable. Set allows you to set a variable to a specified value. Note carefully that if application tasks are running, the value that you write may be overwritten by one of those tasks. Tune allows you to increment or decrement a tunable variable (defined as such in the application task) within its limits. Force allows you to force a variable to a value that is unaffected by any other action in application tasks or the state of physical I/O. Unforce returns a variable to the non-forced state.

Recall that to use any of these options, with the exception of tuning a variable, the keyswitch must be in PROGRAM and the password must be entered. See figure 18.4 for the **Monitor Variable Modify** menu.

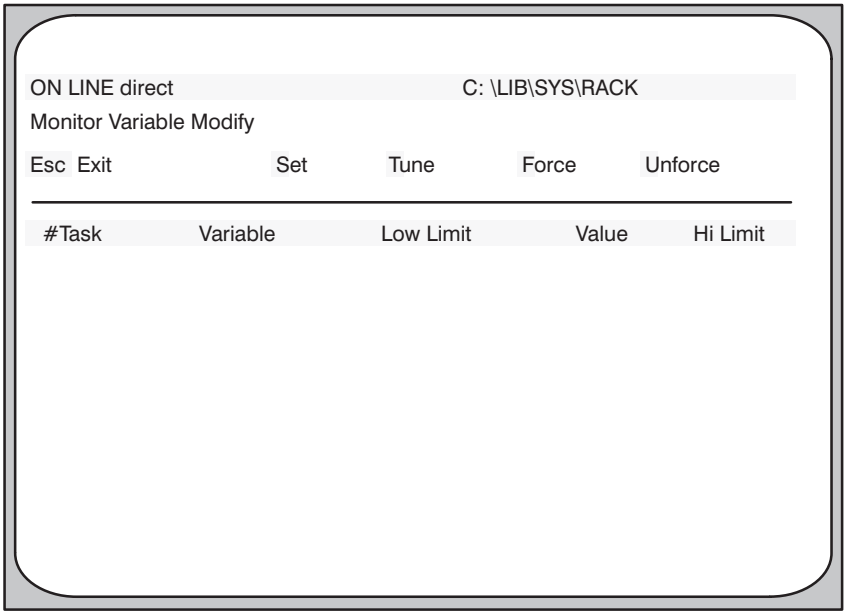


Figure 18.4 - Monitor Variable Modify Menu

18.1.6.1 - 18.1.6.4 below describe the options available from the **Monitor Variable Modify** menu in more detail.

18.1.6.1 Setting a Variable to a Specific Value

You can set variables to specific values using “S” for Set from the **Monitor Variable Modify** menu. The variable you want to set does not need to be in the screen display for you to set it. Note that a variable that has been set may later be affected by an active application task. Follow the directions below to set a variable:

1. Display the variable on the screen following the directions in 18.1.1 above to verify the present value of the variable (optional).
2. Enter “S” for Set from the **Monitor Variable Modify** menu.
3. If the variable you want to set is not displayed on the screen, enter the name of the task where the variable is defined at the blinking cursor, followed by “Enter” or <CR>. Continue at step 4.

   If the name of the variable you want to set is displayed on the screen, you can enter the number of the line (1-16) on which the variable name appears, followed by “Enter” or <CR>. The screen will display the task name in the “Task” field and the variable name in the “Var” field. Skip to Step 5.
4. Enter the name of the variable to be set, specifying variable type with the correct terminator at the blinking cursor, followed by “Enter” or <CR>.
5. The “Value” field will display the last value that was set for a variable. If this is the first time you are setting this variable, the field will be blank. Enter the desired value for the variable, typing

over the last value entered, if it is displayed. The syntax and range of the value you enter will depend on the variable type. See figure 18.3 for the value ranges of variable types. If you are setting a tunable variable, the value you enter must be within the range of the low and high limit.

18.1.6.2 Tuning a Variable

WARNING

DEPENDING ON THE APPLICATION, THE USE OF THE TUNE FUNCTION MAY RESULT IN INSTABILITY OF THE APPLICATION PROCESS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

You can adjust the value of a variable designated as tunable in an application task by selecting "T" for Tune from the **Monitor Variable Modify** menu. The variable you want to tune does not need to be in the screen display for you to tune it. The value is set incrementally between the limits set in the application task. Follow the directions below to tune a variable.

1. Enter "T" for Tune from the **Monitor Variable Modify** menu.
2. If the variable you want to tune is not displayed on the screen, enter the name of the task containing the tunable variable, followed by "Enter" or <CR>. Continue at step 3.

   If the name of the variable you want to tune is displayed on the screen, you can enter the number of the line (1-16) on which the variable name appears, followed by "Enter" or <CR>. The screen will display the task name in the "Task" field and the variable name in the "Var" field. The current value of the variable ("Value" field), high limit ("Hi" field), low limit ("Lo" field), and step value ("Step" field) will also be displayed. Skip to Step 4.
3. Enter the name of the variable, complete with the terminating character that designates variable type, followed by "Enter" or <CR>.

   The current value of the variable ("Value" field), along with the high limit ("Hi" field), low limit ("Lo" field), and step value ("Step" field) will be displayed.
4. Use the up arrow to increment the value in the Value field and the down arrow to decrement the value in the Value field by the Step value.

18.1.6.3 Forcing a Variable

WARNING

THE SET AND FORCE FUNCTIONS BYPASS CONTROL OF THE APPLICATION PROCESS BY THE APPLICATION TASKS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

VARIABLES AND OUTPUTS WHICH ARE FORCED BEFORE AC POWER IS LOST WILL REMAIN FORCED WHEN AC POWER IS RESTORED. SHOULD AC POWER BE LOST WHILE VARIABLES ARE FORCED, THE USER MUST ENSURE THAT UNEXPECTED MACHINE MOVEMENT DOES NOT OCCUR WHEN AC POWER IS RESTORED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

You can force up to 16 variables at a time to specific values. Forcing variables while in the **Monitor Variable Modify** menu is essentially the same as using the Force option described in 18.5. Strings, array elements, and tunable variables cannot be forced. Only booleans, single precision integers, double precision integers, and reals can be forced. If you define bits in a register that is also defined as a register, neither the bits nor the register can be forced. A forced variable cannot be changed in any way by application tasks or the state of physical I/O. The value of a forced variable can only be affected by forcing it to another value or unforcing it. See 18.1.6.4 for more information about unforcing a variable. You can define a reserved common variable, FORCINGSTATUS@, that will indicate whether any variables in the rack have been forced. See 7.2.2 for more information.

Follow the directions below to force a variable.

1. Enter “F” for “Force” from the **Monitor Variable Modify** menu.
2. Enter the name of the task containing the variable to be forced at the blinking cursor. If the variable is a COMMON, you do not need to enter a task name.
3. Enter the name of the variable to be forced, including any terminating character designating variable type, at the blinking cursor.
4. If this is the first time you are forcing a variable, the “Value” field will be blank. Otherwise, the “Value” field will display the last value to which a variable was forced. Enter the desired value for the variable, typing over the last value entered, if it is displayed. The syntax and range of the value you enter will depend on the the variable type. See figure 18.3 for the value ranges of variable types.

18.1.6.4 Unforcing Variables

Unforcing a variable returns it to the state in which it was before it was forced. You can unforce one or all variables that are currently forced. To unforce one or all currently forced variables, follow the steps below.

1. Select “U” for “Unforce” from the **Monitor Variable Modify** menu.
2. To unforce one variable, use the up and down arrows to move the “>” pointer until you reach the desired variable. Then type “Enter” or <CR> to unforce the variable.

To unforce all forced variables displayed on the screen enter “A” for All.

18.2 Monitoring and Modifying I/O

The **Monitor I/O** menu is used to display the state of I/O registers continuously and to modify the state of registers. Registers on the UDC are considered I/O registers. This feature allows you to monitor I/O registers not defined in the configuration for the rack, as well as those that are defined. See figure 18.5 for the **Monitor I/O** menu.

To begin monitoring I/O enter “I” from the **Monitor** menu shown in figure 18.2. The resulting **Monitor I/O** menu, shown in figure 18.5 allows the following options:

Display: Adds an I/O register to the display list. See 18.2.2 for more information.

Clear: Removes a single I/O register or all I/O registers from the display list. See 18.2.3 for more information.

Modify: Allows you to modify an I/O register. See 18.2.7 for more information.

Save : Allows you to save a display list to your default path. See 18.2.4 for more information.

Recall: Allows you to recall a display list that was saved previously. See 18.2.5 for more information.

Locate: Allows you to display the variable name associated with an I/O address or display the I/O address associated with a variable name. See 18.2.6 for more information.

Path: Allows you to change the path.

Files: Allows you to display a directory of the files in the current path.

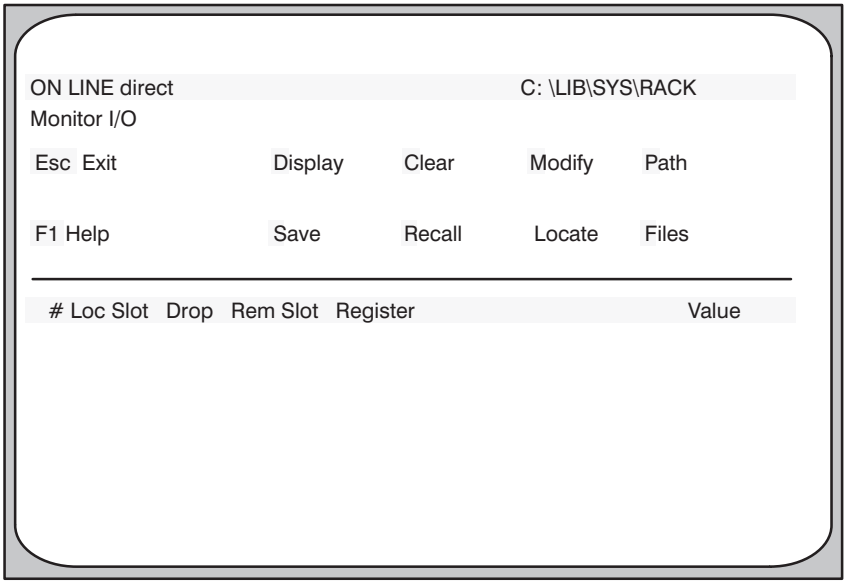


Figure 18.5 - Monitor I/O Menu

18.2.1 Monitor I/O Screen Display

The lower portion of the **Monitor I/O** screen display, which is updated continually, includes the following fields:

- (#) - Display number (screen position).
- Local Slot - The slot (0-15) in the local chassis containing the I/O card to be monitored. If the card to be monitored is in a remote chassis, this is the slot number of the Remote I/O Master to which the remote chassis is connected.
- Drop - The drop number (1-7) of the remote chassis that contains the card to be monitored. If the card to be monitored is in a local chassis, this field is blank.
- Remote Slot - The slot (0-15) in the remote chassis containing the I/O card to be monitored. If the card to be monitored is in a local chassis, this field is blank.
- Register - The register (0-32767) on the I/O card to be monitored.
- Value - The current value of the I/O register. The value is updated in real time. I/O registers may be displayed in either binary format (1s and 0s), decimal format (range: - 32768 to +32767), or hexadecimal format (range: 0 - FFFF).

18.2.2 Displaying I/O Registers

You can display up to 16 registers on the screen at one time. The **Monitor I/O** screen display is described in 18.2.1. To display a register on the screen, follow the directions below:

1. Enter “D” for “Display” from the **Monitor I/O** menu shown in figure 18.5.
2. If the register you want to display is in the local rack, enter “L” for Local and go on to the section entitled “Displaying Local I/O”. If the register is in a remote rack, enter “R” for Remote and go on to the section entitled “Displaying Remote I/O”.

18.2.2.1 Displaying Local I/O

a. Enter the slot number of the module on which the register is found.

b. Enter the register number.

c. Select the format of the display in the “Value” field as follows:
“D” for decimal
“H” for hexadecimal
“B” for binary

d. Type “Enter” or <CR> to display the register.

18.2.2.2 Displaying Remote I/O

a. Enter the slot number of the Remote I/O master module (M/N 57C416) in the local rack.

b. Enter the drop number of the remote chassis in which the module containing the register is located.

c. Enter the slot number of the module in the remote rack that contains the register.

d. Enter the register number.

e. Select the format of the display in the “Value” field as follows:
“D” for decimal
“H” for hexadecimal
“B” for binary

f. Type “Enter” or <CR> to display the register.

18.2.3 Clearing I/O Registers from the Display

To clear I/O from the **Monitor I/O** display, follow the directions in "Clearing a Single Display" or "Clearing the Entire Display," whichever is appropriate. Refer to 18.2.5 for directions on saving a **Monitor I/O** screen if necessary.

18.2.3.1 Clearing a Single Display

1. From the **Monitor I/O** menu, enter "C" for clear.
2. Enter the display position (1-16) of the register you want to clear from the screen.

18.2.3.2 Clearing the Entire Display

1. From the **Monitor I/O** menu, enter "C" for clear.
2. Enter "A" to clear all I/O registers from the screen.

18.2.4 Saving a Monitor I/O Display List

You can save the variables listed on the **Monitor I/O** display to the default path. Only the list of variables is saved, not the variable values. If your default is a floppy disk, it must not be write-protected.

You can give each list a standard 8-character filename. The AutoMax Executive software automatically attaches the file extension ".$IO". You can later recall the list to the screen using the Recall option described in 18.2.6.

To save the current **Monitor I/O** list, follow the directions below:

1. If you have not already done so, enter "S" for Save from the **Monitor I/O** menu shown in figure 18.5.
2. Enter the filename (up to 8 characters) for the display list. Do not attach a file extension. The file will automatically be given the extension $IO.
3. Type "Enter" or <CR>.
4. You can recall the screen display using the Recall option described in 18.2.6.

18.2.5 Recalling a Monitor I/O Display List

You can recall any previously saved **Monitor I/O** display list. To recall the **Monitor I/O** Display list, follow the directions below.

1. From the **Monitor I/O** menu, select "R" for Recall.
2. Enter the name of the display list to recall,
3. Type "Enter" or <CR>.

18.2.6 Locating a Variable or I/O Address

The Locate command can be used to display the variable name associated with an I/O address and display the I/O address associated with a variable name. When you are locating a variable, you have the option of finding the variable using a local I/O address (slot, register) or a remote I/O address (slot, drop, slave slot, register). The information will be retrieved from the configuration file stored in the rack. Note that if you are using interface modules to communicate with foreign I/O, you will not be able to use this command to display Modbus, AutoMate, or A-B register numbers. This command **will** display Modbus, AutoMate, and A-B registers using the equivalent **Multibus** addresses.

Step 1. Select “L” for Locate from the Monitor Variable menu.

Step 2. Select “V” for Variable to locate the variable name. Continue with step 3.

or

Select “I” for I/O to locate the variable’s I/O address. Skip to step 5.

Step 3. Select either “L” for Local I/O Address or “R” for Remote I/O Address.

Step 4. If you selected Local I/O Address, enter the slot, register, and bit number (for boolean variables). If you selected Remote I/O Address, enter the slot, drop, slave slot, register, and bit number (for boolean variables).

The name of the variable that is assigned to the address entered will be displayed.

If there are no variables mapped to the specified I/O point, then the message “There is no variable mapped to this I/O” will be displayed.

Step 5. Enter the variable name.

If the variable is a boolean mapped to an I/O point, the local I/O address (slot, register, bit) and remote I/O address (slot, drop, slave slot, register, bit) will be displayed.

If the variable is an integer, double, or real mapped to an I/O point, the local I/O address (slot, register) and remote I/O address (slot, drop, slave slot, register) will be displayed.

If the variable is not in the rack configuration, the message “Variable is either not found or is a LOCAL variable” will be displayed.

If the variable is mapped to a memory location, the message “Variable is a common memory variable. No I/O address is available” will be displayed.

18.2.7 Modifying I/O Registers

WARNING

MODIFYING I/O REGISTERS MAY AFFECT THE OPERATION OF CONTROLLED MACHINERY. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

The modify option is used to change the value of a register. The register you want to modify does not need to be in the screen display for you to modify it. Recall that if application tasks are running, the value that you write to a register using this option may be overwritten by one of those application tasks.

You must be familiar with how the registers on the module you are working with are organized in order to use this option. Refer to the documentation for the individual module for any questions on register organization.

Follow the directions in 18.2.7.1 for modifying local I/O and in 18.2.7.2 for modifying remote I/O.

18.2.7.1 Modifying Local I/O Registers

WARNING

THIS FUNCTION MAY BYPASS CONTROL OF THE APPLICATION PROCESS BY THE APPLICATION TASKS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

To modify a local I/O register, follow the directions below.

1. If you have not already done so, enter “M” for modify from the **Monitor I/O** menu shown in figure 18.5.
2. Enter “L” for “Local” I/O.
3. Enter the slot number of the module on which the register is found.
4. Enter the register number.
5. Enter the desired value of the register. You may enter a decimal or hexadecimal value only. If you enter a hexadecimal value, the last character of the value must be the letter “H” in upper- or lower-case.
6. Type “Enter” or <CR> to write the value to the register.

18.2.7.2 Modifying Remote I/O Registers

WARNING

THIS FUNCTION MAY BYPASS CONTROL OF THE APPLICATION PROCESS BY THE APPLICATION TASKS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

To modify a remote I/O register, follow the directions below.

1. If you have not already done so, enter “M” for Modify from the **Monitor I/O** menu shown in figure 18.5.
2. Enter “R” for “Remote” I/O.
3. Enter the slot number of the Remote I/O master module (M/N 57C416) in the the local rack.
4. Enter the drop number of the remote chassis in which the module containing the register is located.
5. Enter the slot number of the module in the remote rack that contains the register.
6. Enter the register number.
7. Enter the desired value of the register. You may enter a decimal or hexadecimal value only. If you enter a hexadecimal value, the last character of the value must be the letter “H” in upper- or lower-case.
8. Type “Enter” or <CR> to write the value to the register.

18.3 Outputting Analog Signals

If your system configuration incorporates a DCS 5000 micro-regulator drive, the **Monitor Analog** menu allows you to output two analog signals that are proportional to the values of the specified integer variables. These signals can be used to drive a chart recorder, oscilloscope, or other test instrument. The **Monitor Analog** is functional only with a Processor containing a current minor loop (CML) application task, i.e., a task containing the CML control block. See J-3676 for more information on the Control Block language.

The analog signals are obtained from the the D/A converters located on the Drive Analog I/O Module (B/M 57405). The variable values are scaled and updated once every clock tick as long as the CML task is running. The D/A voltage is:

D/A(volts) = variable * scale *(10 volts/4096 counts)

A maximum of two single precision integer variables can be displayed simultaneously for each CML task, one per channel. Each variable is identified by the D/A channel from which the signal is obtained. The lower channel is identified as 0, and the upper channel is identified as 1.

18.3.1 Monitor Analog Display

The following fields will be displayed on the **Monitor Analog** screen for each variable:

(#)	-	Identifies the D/A channel (0 or 1) through which the value will be displayed.
Task	-	Identifies the task in which the variable is defined. This field is blank if the variable is common.
Variable	-	Variable name, including the type character "%" for single precision integers.
Scale	-	Identifies a scale factor of *16, *8, *4, *2, *1, *1/2, *1/4, *1/8, or *1/16.

18.3.2 Outputting a Variable

The procedure for adding a variable is the same whether you are actually adding the variable, or replacing a variable with another variable.

Follow the procedure below to output a variable.

1. From the **Monitor** menu, select "A" for analog.
2. Enter the slot number of the Processor on which the CML task is running.
3. Select "A" for Add to add a variable.
4. Enter the D/A converter through which the signal is obtained (0 or 1). The converter corresponds to the display position. If you enter a display position currently on the screen, the variable associated with it will be replaced by the new variable you enter in step 5 below.
5. If the variable in question is a local variable, enter the name of the task in which the variable is defined. If the variable is a common variable, i.e., configured for the rack using the Variable Configurator, press the "Enter" key or <CR> to skip over the "Task" field.
6. Enter the name of the variable in the "Variable" field. Include the type character "%" for single precision integers. Type "Enter" or <CR>.
7. When the cursor has moved to the "Scale" field, use the up and down arrow keys to scroll through the available scale factors described in 18.3.1 above. The default scale factor is 1.
8. Press "Enter" or <CR> to terminate the entry and add it to the display.

18.4 Monitoring and Editing PC Tasks

WARNING

USE OF THE ONLINE EDITING FEATURE CAN CAUSE THE CONTROLLED PROCESS OR MACHINE TO SUDDENLY CHANGE ITS OPERATION. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

Before you change any Ladder Logic task online, you should be familiar with the information in 4.22.1, which describes variable types and naming conventions in AutoMax Ladder Logic tasks.

The **Monitor PC** menu is used to monitor the state of Ladder Logic task sequences in real-time and – unlike file editing – to modify individual control logic sequences **online while tasks are running**. To make any changes to Ladder Logic tasks, the keyswitch must be in PROGRAM and the password must be entered.

Online editing operations supported include adding, deleting, and modifying an existing sequence, modifying the value of a preset, forcing/unforcing a variable, moving one or a set of sequences, and resequencing all or part of a task. You can only modify one sequence (edit one sequence, add one sequence, or delete one sequence) at a time. Any online changes made take effect during the scan immediately following the task scan during which the change is installed.

When you monitor a sequence, the power flow is indicated by reverse video display for those portions of the sequence that have electrical continuity. If a sequence contains a timer or counter, the current value is displayed, along with the preset value. For shift registers, the current contents of the register are displayed. Variable names displayed in lower-case indicate local variables. Those displayed in upper-case are common variables. See figure 18.6 for a sample display.

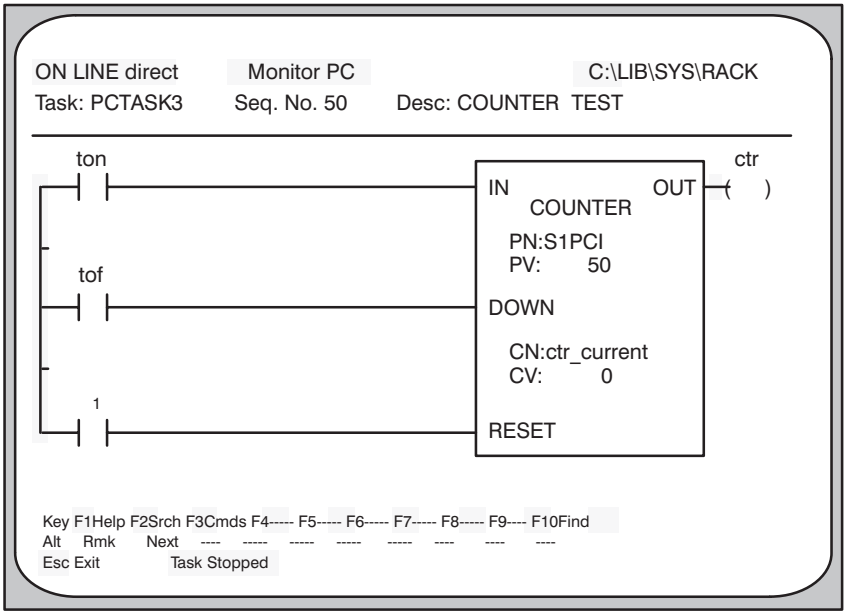


Figure 18.6 - Sample Monitor PC Display

While monitoring a sequence, you can use the four arrow keys to position the cursor over any element in the sequence. If there is a description for the selected element, it will be displayed on the screen in the field “Desc”.

While you are in the **PC Monitor**, the following keys have the special functions indicated.

Home	Displays the first sequence in the task
PgUp	Displays the previous sequence in the task
PgDn	Displays the next sequence in the task
End	Displays the last sequence in the task

18.4.1 PC Monitor Online Edit Commands

WARNING

USE OF THE ONLINE EDITING FEATURE CAN CAUSE THE CONTROLLED PROCESS OR MACHINE TO SUDDENLY CHANGE ITS OPERATION. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

F3 is used to initiate an online edit command. F3 followed by one of the following keys executes the indicated command. Each command is described in the section listed to its right.

Ins	- Selects INSERT SEQUENCE	See 18.4.2 and 18.4.3
Del	- Selects DELETE SEQUENCE	See 18.4.4
C	- Selects CHANGE SEQUENCE	See 18.4.2 and 18.4.5
P	- Selects the MODIFY PRESET menu	See 18.4.6
F	- Selects the FORCE menu	See 18.4.7
U	- Selects the UNFORCE menu	See 18.4.8
R	- Selects the RESEQUENCE menu	See 18.4.9
M	- Selects the MOVE SEQUENCE menu	See 18.4.10

When you initiate an online edit command as described below in 18.4.3 and 18.4.5, the procedures will be essentially the same as insertion and editing in the the offline PC Editor described in 4.22.2 - 4.22.5, with the following exception: F9 R may not be used to create a new remark sequence with the **PC Monitor**.

18.4.2 Limitations on Inserting and Modifying Sequences

There are a number of limitations on inserting and modifying a sequence or sequences into a Ladder Logic task that is running. These are detailed individually below.

18.4.2.1 Number of Sequence Insertions

When a PC task is downloaded, 1024 bytes are reserved for changes to PC task runtime code. Once this is used up, the screen will display an error message indicating that the memory is full. The smallest possible sequence uses 14 bytes of memory. The smallest possible block uses 22 bytes.

18.4.2.2 Adding Variables

From the time a task is downloaded, only 16 new symbols may be added to the task. They may be any combination of local booleans, common booleans, local integers, and common integers, but when the total number of new symbols exceeds 16, the screen will display an error message indicating the symbol table is full. Deleting a sequence does not remove symbols from the symbol table. In other words, if you add a sequence with two new symbols and then later take that sequence out, you will still have used up two of the 16 allowable new symbols.

18.4.2.3 Using an Existing Coil Name

You cannot insert a sequence with the same coil name as that found in an existing sequence. If you try to do so, the system will display an error message indicating there is a duplicate coil.

18.4.2.4 Adding Local Booleans

All local booleans that are added must first be entered as coils in order to be defined as locals. They must be defined as locals in this way before they may be used as contacts. All common symbols that are added must be defined using the Variable Configurator. If a new common that is not already defined is added, the screen will display an error message indicating that the symbol is not resolved.

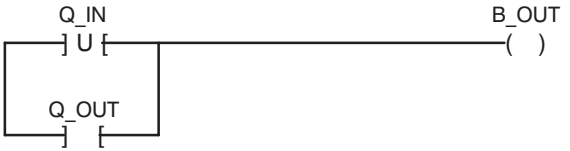
18.4.2.5 Inserting a Sequence with Counter or Shift Register Blocks

If Counter or Shift Register blocks are installed when the UP, DOWN, or SHIFT inputs are conducting, the count or shift will not take place. The count or shift will take place only when the input is turned off and back on again.

18.4.2.6 Inserting or Modifying a Sequence with an Upward Transition Contact

If an upward transition contact references a new input in a register not previously used by the task, and if the input is conducting, an upward transition will be detected on the first scan after the new sequence is installed or the existing sequence modified.

For example, you insert the following sequence where Q_IN is defined in the configuration task for the rack, but is not used elsewhere in the Ladder Logic task:



If Q_IN is on when the sequence is inserted, B_OUT will turn on. If a different bit in the word where Q_IN is defined was already used by the task, then B_OUT will not turn on until Q_IN turns off and then on again.

18.4.3 Inserting a Sequence

WARNING

USE OF THE ONLINE EDITING FEATURE CAN CAUSE THE CONTROLLED PROCESS OR MACHINE TO SUDDENLY CHANGE ITS OPERATION. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

Make certain you have read and understood the restrictions in 18.4.2.1 - 18.4.2.5 before proceeding. Recall that when you are inserting a sequence while in the **PC Monitor**, the procedure will be essentially the same as it is when you are creating the task in the offline PC Editor described in 4.22.2 - 4.22.5, with the exceptions described in 18.4.1.

To insert a sequence, follow the directions below.

1. Monitor the sequence that will precede where you want the new sequence to be inserted. For example, if you want to insert a sequence between existing sequences 20 and 30, you begin by monitoring sequence 20. The insertion will always occur immediately following the sequence being monitored. To insert a new sequence before the first sequence in the task, see 18.4.2.
2. Once you are monitoring the correct sequence, enter F3 followed by the “Ins” key. The system will test whether there is a sequence number that will fit between the monitored sequence and the one following it. For example, if the line number of the sequence being monitored is a multiple of 10, the system will test whether that line plus 10 is available. In the above example, where there is already a sequence 30, the system would then test if the monitored sequence line number plus one is available. If a sequence 21 already existed, you would be prompted to resequence the task. See 18.4.9 below for more information about resequencing.
3. After a valid sequence number is assigned for the sequence to be added, the sequence being monitored is cleared from the screen to allow you to enter the new sequence. For help in inserting the sequence, type F1.
4. After you have finished entering the new sequence, type F3 E to exit the editor and install the new sequence. If you change your mind and no longer want to insert the sequence, enter F3 Q. If you type F3 by mistake and you want to go back and edit the sequence further, type ESC.

18.4.4 Deleting a Sequence

WARNING

USE OF THE ONLINE EDITING FEATURE CAN CAUSE THE CONTROLLED PROCESS OR MACHINE TO SUDDENLY CHANGE ITS OPERATION. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

To delete a sequence, follow the directions below.

1. Monitor the sequence that you want to delete.
2. Enter F3 followed by the Del key. The system will prompt with "Are you sure [n]?" If you enter anything other than a Y for yes, the sequence will not be deleted. All symbols that are used by the sequence remain in the symbol table. The 16 symbol limit described in 18.4.2.2 is not affected by deleting a sequence.

18.4.5 Editing a Sequence

WARNING

USE OF THE ONLINE EDITING FEATURE CAN CAUSE THE CONTROLLED PROCESS OR MACHINE TO SUDDENLY CHANGE ITS OPERATION. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

Make certain you have read and understood the restrictions in 18.4.2.1 - 18.4.2.5 before proceeding. Recall that when you are editing a sequence while in the **PC Monitor**, the procedure will be essentially the same as it is when you are creating the task in the offline PC Editor described in 4.22.2 - 4.22.5, with the exceptions described in 18.4.1.

To edit a sequence, follow the directions below.

1. Monitor the sequence you want to edit.
2. Enter F3 followed by "C" for Change.
3. For help in editing the sequence, enter F1.
4. After you have finished editing the sequence, type F3 E to exit the editor and install the edited sequence. If you change your mind and no longer want to insert the sequence, enter F3 Q. If you type F3 by mistake and you want to go back and edit the sequence further, type ESC.

18.4.6 Modifying a Preset

To modify the preset value for a COUNTER or TIME_ON or TIME_OFF block, follow the directions below. You cannot use this function if the preset is currently forced.

1. Enter F3, followed by “P” for Preset.
2. Enter the name of the task in which the preset is to be modified.
3. At the next prompt, enter the name of the preset followed by a % sign. The name you enter must be an existing integer name or an error message will be displayed.
4. Enter the new value. The new value is written into the variable and is saved as a part of the task. The unit is 1/10 of a second. The new value must be in the range of 1 to 32767 for timers and −32768 to +32767 for counters.

18.4.7 Forcing a Variable

WARNING

THE SET AND FORCE FUNCTIONS BYPASS CONTROL OF THE APPLICATION PROCESS BY THE APPLICATION TASKS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

You can access the same functions allowed in the Force menu described in 18.5 to force a variable while in the **PC Monitor**. Strings, array elements, and tunable variables cannot be forced. Only booleans, single precision integers, double precision integers, and reals can be forced. Once a variable is forced, its value cannot be altered by application tasks or the status of physical I/O. Its value can only be changed by forcing it to another value or unforcing it. See 18.4.8 for more information about unforcing a variable. To force a variable, follow the directions below.

1. Enter F3, followed by “F” for Force.
2. Enter the name of the task in which the variable is to be forced. If the variable is common, skip over this step.
3. Enter the name of the variable. The variable name must be terminated by the @ character for boolean variables, the % character for integer variables, or the ! character for double integer variables. Forced boolean variables are shown blinking on the display.
4. Enter the value to which the variable is to be forced in the “Value” field.
5. You can enter further values for the variable without entering F3 F if you have not exited using the <Esc> key. Enter the new value over the old.
6. To exit, type <ESC>.
7. To unforce a variable or all variables, follow the directions in 18.4.8.

18.4.8 Unforcing a Variable

WARNING

THE SET AND FORCE FUNCTIONS BYPASS CONTROL OF THE APPLICATION PROCESS BY THE APPLICATION TASKS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

You can unforce variables by removing them from the Force table.

To remove variables from the Force table, follow the directions below.

1. Enter F3 followed by “U” for “Unforce”.
2. To unforce all variables in the Force table, enter an “A”. To unforce one variable, use the up and down arrow keys to move the “>” pointer until the desired variable is displayed and then type a <CR>. If the force table is emptied, the system will exit to the menu.
3. To exit the function, type <ESC>.

18.4.9 Resequencing a Task

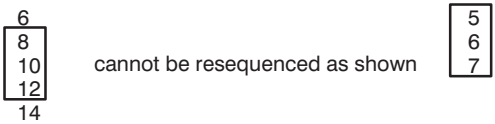
The method used to resequence tasks in the **PC Monitor** is the same as the method used in the PC Editor, except for the fact that tasks resequenced in the PC Monitor are running and the change takes effect with the next scan.

To resequence a task, follow the directions below.

1. Type F3 followed by “R” for Resequence.
2. At the prompt “Old First Sequence”, enter the sequence number at which you want to begin the resequence.
3. At the prompt “Old Last Sequence”, enter the upper limit for the range of sequences you want to resequence. The range you select is inclusive.
4. At the prompt “New First Sequence”, enter the new sequence number for the first sequence in the range.
5. At the prompt “New Increment”, enter the number to add to subsequent sequence numbers, beginning with the “New First Sequence”.

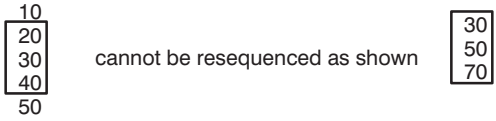
The following are not permitted when you resequence and will cause the system to display an error message if attempted:

* If “New First Sequence” is smaller than the sequence immediately preceding “Old First Sequence”. For example, you cannot resequence the sequences shown boxed on the left as those shown boxed on the right:



* If the new last sequence in the range would become greater than the sequence that follows the “Old Last Sequence”. For example,

you cannot resequence the sequences shown boxed on the left as those shown boxed on the right:



* If the Old First Sequence or Old Last Sequence is not a sequence in the task, the system will display an error message indicating that the sequence was not found.

18.4.10 Moving a Sequence or Set of Sequences

The method used to move a sequence or set of sequences in the **PC Monitor** is the same as the method used in the offline PC Editor described in 4.22.2, except for the fact that tasks edited in the **PC Monitor** are running and the change takes effect with the next scan. To move a sequence or set of sequences, follow the directions below.

1. Enter F3 followed by “M” for “Move”.
2. At the prompt “First Sequence To Move”, enter the sequence or first of the set of sequences to be moved.
3. At the prompt “Last Sequence To Move”, enter the last sequence or last of the set of sequences to be moved. If you are moving only one sequence, enter the same sequence number as you did in number 2 above. If you are moving a set of sequences, the range of sequences between the first and last sequence numbers you enter is inclusive.
4. At the prompt “New Sequence Number”, enter the new number of the sequence or the first sequence of the set. If you are moving a set of sequences, they will be incremented by 10 after the first sequence.

The following are not permitted when you move a sequence and would cause the system to display an error message if attempted:

* If the sequences starting with the new sequence number would fall into the same range as existing sequences.
* If either the “First Sequence to Move” or “Last Sequence to Move” is not a sequence in the task.

18.4.11 Searching for a Sequence Number or Coil Variable Name

You can search a task for a sequence number or coil variable name using F10. To find a particular sequence number or coil variable name, follow the directions below.

1. Enter F10.
2. At the prompt, enter either the sequence number or the name of the coil. That sequence will then be displayed on the screen.

18.4.12 Searching for a Variable Name

You can search a task for a variable by following the directions below.

1. Enter F2 to initiate the search.
2. Enter the name of the variable.
3. By entering one of the letters indicated below, you can qualify the search to examine only certain types of elements for the variable:

   A - All contacts and coil
   O - Normally open contacts
   C - Normally closed contacts
   U - Upward transition contacts
   D - Downward transition contacts

   The task will be scanned from the beginning for a match. If a match is found, the search stops and the ladder sequence containing the match is displayed.
4. If you want to search for another instance of the same variable, enter Alt/F2. This function will cause the search to continue from the sequence at which it had stopped until the next match. A search must have been initiated with the F2 function key in order to use this function. It may be re-executed as many times as necessary.

18.4.13 Inserting a New First Sequence

To insert a sequence before the existing first sequence, follow the directions below. This is the sole method of inserting a sequence before an existing first sequence. F10 can also be used to insert at any place in the task if the sequence number entered is not already in the task.

1. Enter F10.
2. Enter a sequence number smaller than the current first sequence number.
3. Proceed with the insert operation described in 18.4.2 and 18.4.3 above.

18.4.14 Remark Sequence Text

You can use Alt/F1 to toggle on/off the display of Remark sequence text.

18.5 Forcing Variables

WARNING

THE SET AND FORCE FUNCTIONS BYPASS CONTROL OF THE APPLICATION PROCESS BY THE APPLICATION TASKS. IT IS THE RESPONSIBILITY OF THE USER TO DETERMINE THE POTENTIAL HAZARDS INVOLVED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

WARNING

VARIABLES AND OUTPUTS WHICH ARE FORCED BEFORE AC POWER IS LOST WILL REMAIN FORCED WHEN AC POWER IS RESTORED. SHOULD AC POWER BE LOST WHILE VARIABLES ARE FORCED, THE USER MUST ENSURE THAT UNEXPECTED MACHINE MOVEMENT DOES NOT OCCUR WHEN AC POWER IS RESTORED. FAILURE TO OBSERVE THESE PRECAUTIONS COULD RESULT IN BODILY INJURY.

From the **Monitor** menu shown in figure 18.1, you can select “F” for Force to force and unforce up to 16 variables at the same time. Note that the limit of 16 forced variables applies regardless of the number of users who have access to the tack. You can only force boolean, single precision integer, double precision integer, and real variables. Strings, array elements, and tunable variables cannot not be forced. A variable that is forced cannot be altered by any application task or the status of physical I/O. A forced variable will retain its forced value until it is unforced or forced to a different value.

See 18.5.1 for an explanation of the screen display and 18.5.2 and 18.5.3 for instructions on forcing and unforcing variables.

18.5.1 Monitor Force Screen Display

Forced variables are displayed in a table on the screen. The columns in the force table contain the following information:

(#)	-	Force table (screen) position (1-16).
Task	-	The task in which the variable is defined. This field is blank if the variable is COMMON.
Variable	-	Name of the variable being forced, including the type character.
Value	-	The current forced value of the variable.

See figure 18.7 for a the **Monitor Force** menu.

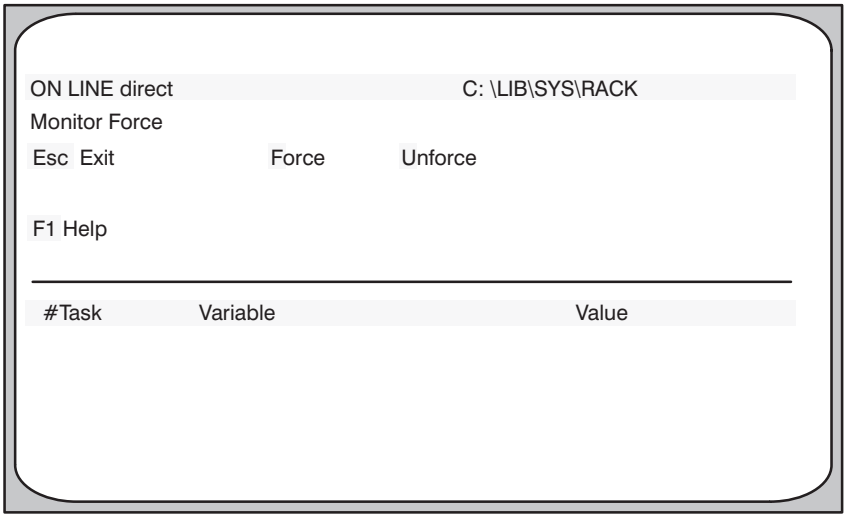


Figure 18.7 - Monitor Force Menu

18.5.2 Forcing a Variable

To force a variable, follow the directions below.

1. At the **Monitor** menu, enter "F" for Force. See Figure 18.8 for a sample screen containing 14 forced variables. Note that no task is shown for common variables.

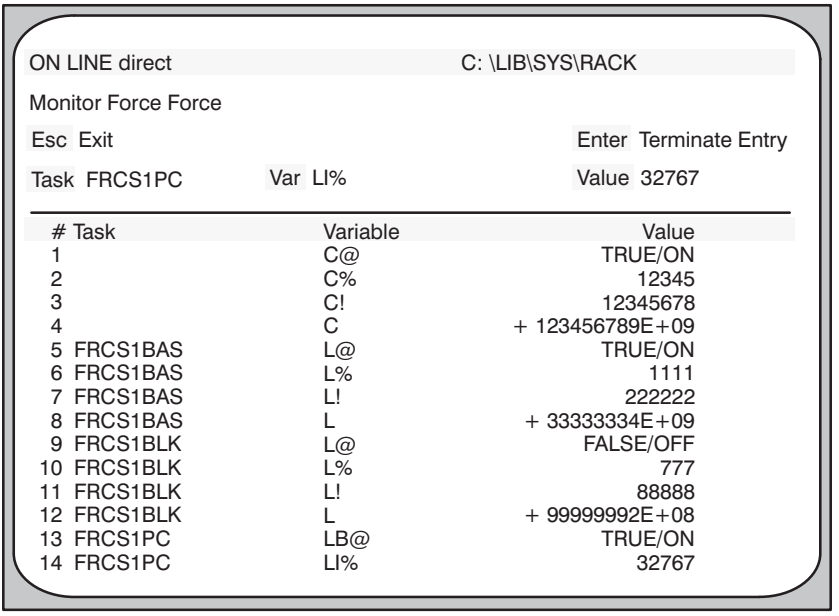


Figure 18.8 - Sample Monitor Force Display

2. Enter the name of the task in which the variable is to be forced.
3. Enter the name of the variable, terminated by the appropriate character (“@” for boolean, “%” for single precision integers, or “!” for double precision integers).
4. Enter the value to which the variable is to be forced in the “Value” field.
5. You can enter different values for the same variable if you have not exited using the <Esc> key. Simply use the arrow keys to move the cursor to the “Value” field again and enter the new value over the old.
6. To exit, type <ESC>.
7. To unforce one variable or all variables, follow the directions in 18.5.3.

18.5.3 Unforcing a Variable

You can unforce variables by removing them from the Force table. To remove variables from the Force table, follow the directions below.

1. Enter “U” for “Unforce”.
2. To unforce all variables in the Force table, enter an “A”. To unforce one variable enter the display position of the variable to be unforced. If the force table is emptied, the system will exit to the **Monitor** menu.
3. To exit the function, type <ESC>.

18.6 Setting Up the UDC and PMI Meter Ports

If your system configuration incorporates an AutoMax Distributed Power drive, the **Monitor Set Up UDC/PMI** menu allows you to select variables to drive the four D/A output ports (labeled “Meter Ports”) on each of the UDC modules and PMI Processors. The analog signals output by these ports can be used to drive meters, chart recorders, or other test instruments. Refer to the appropriate DPS Configuration and Programming instruction manual for more information on connecting equipment to the UDC and PMI meter ports.

After the slot number of a UDC module is entered, the user can enter or modify setup information about the meter ports for the UDC module or for either of the PMI Processors associated with that UDC module. See figure 18.9 for the UDC Setup screen display.

Note that both the configuration file and the drive parameter file (.POB file) must be downloaded to the UDC before you can set up the UDC or PMI meter ports using Monitor.

Other Methods of Setting UP Meter Ports

PMI meter ports can be set up during DPS parameter entry on on the Monitor Setup UDC/PMI menu. UDC meter ports can be set up via registers 1001−1017 in a UDC application task or on the Monitor Setup UDC/PMI menu. The setup information may be retained after a Stop All or power cycle depending on the method used. See the table below.

Table 18.1 − Meter Port Setup Retention

	UDC Setup via Monitor Menu	UDC Setup via Application Task (Reg. 1001- 1017)	PMI Setup during Parameter Entry	PMI Setup via Monitor Menu
Retained after Stop All?	Yes	Yes*	Yes	No**
Retained after power cycle	No	Yes*	Yes	No**

*Original task setup is retained.

**If the PMI meter ports are set up using the parameter screens and then later set up again using the Monitor menu, the setup in the parameter screens will still be retained.

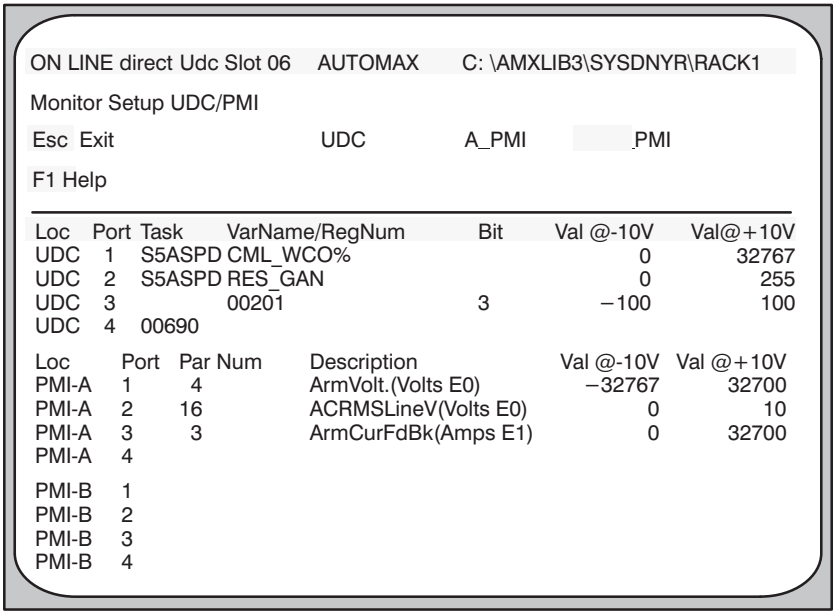


Figure 18.9 - UDC Setup Screen

You can display and output to the selected port any variable mapped to a UDC register (common) or any local variable. The task name is required for local variables.

Each UDC entry has a port number (1-4), a variable name or register/bit number, and a minimum (Val @ −10V) and maximum (Val @ +10V) value. “Val @ −10V” specifies the value of the variable that will be output as the minimum voltage, and “Val @ +10V” specifies the value of the variable that will be output as the maximum voltage. This allows a small area of the signal range to be displayed over the full range of the 8-bit digital-to-analog converter. Values are transferred to the analog output every scan if CCLK has been turned on in a UDC task that is running, and every 5 milliseconds if CCLK is off.

Each PMI entry has a port number (1-4), a parameter number and description, and a minimum and maximum value. You can select the parameter from a list of available parameters and set the “Val @ −10V” and “Val @ +10V” values in the same way as for a UDC module. Only data that exists on the selected PMI Processor can be output to one of its meter ports. Values are transferred to the analog output after every current minor loop scan in the PMI Processor.

Use the following procedure to set up the meter ports for the UDC modules and PMI Processors.

Step 1. Enter “S” for Set Up UDC/PMI from the **Monitor** menu shown in figure 18.1.

Step 2. Enter the slot number of the UDC module. The screen will display the current variable name (or register and bit number) and the minimum and maximum values for each port on the UDC module. It will also display the parameter number and name and the minimum and maximum values for each port on the PMI Processors associated with the selected UDC module (A and B).

Step 3. Enter “U” to select the UDC analog outputs, “A” to select the PMI-A analog outputs, or “B” to select the PMI-B analog outputs.

Step 4. Enter the port (1-4) you want to configure.

Step 5. If you entered “U” (UDC) above, enter the task name (for LOCAL variable), the variable name or register and bit number of the variable you want to assign to the selected UDC meter port.

If you selected “A” (PMI-A) or “B” (PMI-B), select the parameter you want to assign to the port. Use the Up Arrow and Down Arrow keys to scroll through the list of available parameters displayed on the screen.

Step 6. Enter a minimum and maximum value for the selected variable in the “Val @ −10V” and “Val @ +10V” fields, respectively.

Step 7. When you are finished setting up the UDC and PMI meter ports, press <ESC> to return to the Monitor menu.

19.0 ON LINE MENU: ERROR CLEAR

Any errors in the AutoMax Processor error log, the UDC error log, or on the AutoMax Processor LEDs should first be cleared if you wish to run any tasks on that particular Processor or UDC module. The Error Clear menu allows you to clear the error logs of AutoMax Processor modules and UDC modules as well as the LED fault codes on AutoMax Processor modules. This option simply clears the error log and the LEDs; it does not correct the errors themselves. Follow the steps below to clear a UDC error log, an AutoMax Processor error log, the LEDs on an AutoMax Processor, or the error log and LEDs for an AutoMax Processor.

1. From the ON LINE menu shown in figure 11.1, select “E” for Error Clear.
2. Enter the slot number of the AutoMax Processor or UDC module whose error log you want to clear.
3. If you entered the slot number of a UDC module, select “E” to clear the error log.

If you entered the slot number of an AutoMax Processor, select “E” to clear the error log, “L” to clear the LEDs, or “B” to clear both the error log and LEDs.

20.0 SOFTWARE TROUBLESHOOTING

Error messages may be displayed on the screen if there are problems with memory, disks, or databases. This section describes some actions you can take to correct these problems.

20.1 Memory Errors

Some errors are caused by insufficient memory. An error message will indicate if this occurs. The Executive software requires 1MB of RAM. When one of these errors occurs, exit the Executive, make more RAM memory available and re-start the application.

Memory can be freed by:

- closing other Windows applications
- removing any terminate and stay resident (TSR) programs (e.g., Sidekick)
- removing any network drivers or moving them to high memory
- removing any other drivers or moving them to high memory
- reducing the memory reserved for environment variables
- reducing the number of files and buffers reserved in CONFIG.SYS (Windows requires at least 30 files and 10 buffers).

The *Microsoft Windows User's Guide* provides additional information on configuring your personal computer's memory.

20.2 Disk Errors

Some errors are caused by problems with the disk on which files are stored. An error message will indicate if this occurs. When one of these errors occurs, check for problems with the disk or disk drive and re-start the application.

Disk problems include:

- a write-protected disk
- lack of sufficient disk space
- protection on the disk or directory (for a drive on a network server)
- hardware problems with the disk drive

20.3 Database Problems

Errors can be caused by inconsistent, invalid, or missing data and/or index files. An error message will indicate if this occurs.

One source of problems with the data and index files is changes made outside the AutoMax Programming Executive software. These changes can be made with DOS (e.g., renaming or deleting files) or using a dBASE (or compatible) software package to write to the file. If any of these changes were made, verify that the changes were made correctly and that the index files are current.

Recovering the data and index files from a valid backup copy should correct the problem. In addition, it is possible to repair data and index files using a dBASE (or compatible) software package. However, this requires an understanding of dBASE and of the databases used in the AutoMax package. See section 4.19 and Appendix F for information on the AutoMax database organization.

If the application task files for a rack are deleted or become corrupted, you may be able to recover them by saving them from the Processor back to your personal computer. The procedure that follows can recover only the reconstructible application tasks.

Step 1. From the System Configurator, create a temporary rack (e.g., TEMP) in the desired system and section by using the Add command from the Rack menu (See section 5.1.3).

Step 2. Access the Task Manager and go online by selecting the On Line command (See section 8.17).

Step 3. Enter “S” (Save) at the Transfer menu to bring up a list of all the tasks in the rack. Write down the task names. You will need these names later in order to recover the utility tasks (See section 14.4).

Step 4. Enter the name of the rack configuration object file (_CONF.CNF) and use the /All option to save all the tasks in the rack to TEMP (See section 14.4). Then press F3 to exit the On Line menu and return to the Task Manager.

Step 5. Access the System Configurator. Use the Import command from the Rack menu to import the tasks in TEMP to the new rack. Enter the appropriate Disk:\library\system\TEMP in the Source Directory for Imported Rack dialog box (See section 5.3.8).

Step 6. Access the Task Manager. Select Add from the Task menu and use the New option to copy the utility tasks to the task list (See section 8.1.1). All of the application tasks should now be listed for the rack.

Step 7. Select Generate Configuration from the Commands menu to re-create the configuration object file for the rack (See section 8.16.3).

Step 8. Return to the System Configurator and use the Remove command from the Rack menu to delete the temporary rack (See section 5.3.3).

20.4 Module/Variable Compatibility Problems

If registers defined in an AutoMax V2 module cannot be configured in the current version of the Executive software, or the database is somehow corrupted, an error message will be displayed (invalid/incompatible data type) when you try to configure the module. This error can occur because the current Executive software treats duplicate status registers, e.g. “Interrupt Enable,” differently than in version 2. If one register is read only (R) and the other read/write (R/W), the current software allows you to configure just the read only register. Note that for duplicate status registers, R/W refers to the operating system’s ability to write to that register, not the application task’s. An error message will be displayed for each of the affected registers.

To fix this error, write down the variable name and address displayed, then delete the variable. You will not be able to configure the module unless you delete the offending variable. When you have deleted all offending variables, configure the module as desired. You will have to edit your application tasks to reflect the new configuration.

20.5 Printing Problems

The Executive software will print all tasks and files 132 characters per line. It does not, however, change page orientation and will not use soft fonts. Printers must be set up in the Windows Setup procedure (see section 5.5.1) and the AutoMax Printer Setup procedure (see section 5.5.4).

If you experience problems with printing AutoMax files, try printing a file from the Windows Notepad to determine if the problem is with the Executive software or with the Printers Setup. See the *Microsoft Windows User’s Guide* for additional information.

20.6 System Errors

Errors that indicate a problem with the Executive software will cause the application in which they occur to close. If this type of error occurs, write down the error message; then contact Reliance.

Appendix A

Processor Module Ports Programmer/Port B

If the personal computer that will be used as a programming terminal for the Processor module was purchased from Reliance, you will receive with the computer the proper cable to connect the computer and the Processor module.

WARNING

THE FOLLOWING INSTRUCTIONS ARE INTENDED ONLY TO ALLOW FABRICATION OF PROPER CONNECTIONS BETWEEN RELIANCE EQUIPMENT AND USER-PROVIDED PROGRAMMING DEVICES. THE USER MUST READ AND UNDERSTAND ALL APPLICABLE INSTRUCTION MANUALS PRIOR TO OPERATING THE EQUIPMENT. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY.

If it is necessary to prepare a cable to connect a programming terminal to the RS-232 port labeled "Programmer/Port B" on the leftmost Processor in the AutoMax rack, follow the steps outlined below.

1. Determine whether your programming terminal contains a 9- or 25-pin male connector.
2. Cut a suitable length (not to exceed 10 feet) of 22-gauge, multi-conductor cable.
3. Follow the connector manufacturer's instructions and make cable connections using figure 1 or 2, whichever is applicable.
4. Check for grounds, shorts, and continuity using an Ohm meter.
5. Read and understand all applicable Reliance instruction manuals before connecting the cables to Reliance equipment.

Programming Terminal End 25-Pin Female Connector			Reliance End 25-Pin Male Connector	
SIGNAL	PIN#		PIN#	SIGNAL
RECV	3	←	2	XMIT
XMIT	2	→	3	RECV
CTS	5	←	4	RTS
RTS	4	→	5	CTS
DTR	20	→	6	DSR
DSR	6	←	20	DTR
COM	7	↔	7	COM

Figure 1

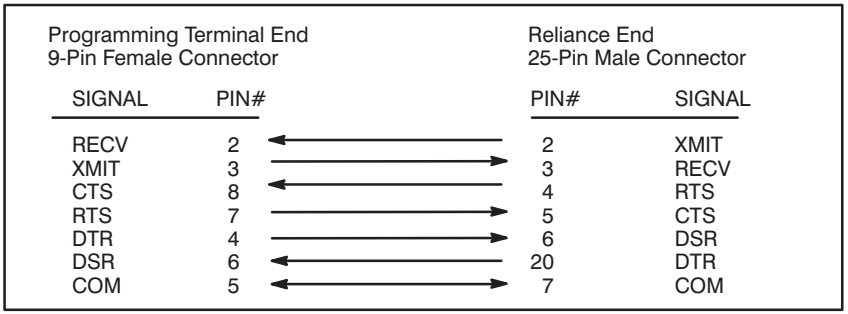


Figure 2

User Serial Ports

All Processor module ports except for the port “PROGRAMMER/PORT B” on the leftmost Processor in the rack are available to the user to connect to an external device. In order for the application software to be able to use a serial port, you must first use the OPEN statement in a BASIC task. Refer to the Enhanced BASIC Language Instruction Manual (J-3675) for more information. Note that on AutoMax Processor modules, you can OPEN “PORTA” or OPEN “PORTB”. The following is a description of the pins on ports available to the user.

Pin#	Function
2	Output. This signal contains transmitted data.
3	Input. This signal contains received data.
4	Output. This signal indicates transmitter status and is true whenever the transmitter is sending characters. The signal is typically used to enable an external device, e.g.,a modem. The signal is meaningful only if hardware handshaking has been enabled.
5	Input. This signal enables the transmitter. It must be true in order for the transmitter to send a character. The signal is typically used for hardware flow control. This input is ignored if hardware handshaking has not been enabled.
6	Input. This signal enables the receiver. It must be true in order for the receiver to accept characters. If this signal becomes false while a message is being received, any characters being received will be deleted and an error will be reported to the application software. This input is ignored to the application software. This input is ignored if hardware handshaking has not been enabled.
7	Signal ground.
20	Output. This signal indicates receive status. The signal is true whenever the receiver will accept characters. It is typically used for hardware flow control. The signal is meaningful only if hardware handshaking has been enabled.

Appendix B

Processor Module Error and Status Codes

All AutoMax Processor module error and status codes (displayed on the two seven-segment LEDs on the faceplate) are listed below, organized first by error type and second, in numerical/alphabetical order.

Processor Overload

00	CPU overload

Corrective action: move one or more application tasks to other Processor modules in the rack, or make scan times longer.

Power-Up Diagnostics

The following error codes are displayed while the Processor module performs power-up diagnostics.

0.0.	EPROM failed
0.1.- 0.3.	Bad CPU
0.4.	Internal bus error test failure
0.5.	Parity test failure
0.6.	External bus error test failure
0.7.	Processor in the wrong slot
1.0. - 1.6.	RAM failure
2.0.	I/O protection failure
2.1.	PIO failed
2.2.	PC accelerator failed
2.3.	8253 timer/counter failed
2.4.	SIO failure
2.5.	Communications interrupt failed
2.6.	SIO interrupt failed
2.7.	8253 counter/timer interrupt failed
2.8.	Local watchdog failed
3.0.	Bad backplane
3.1.	Multibus parity test failure
4.0. - 4.5.	Common memory RAM failure
4.6.	Common memory system watchdog failure
5.0.	Processors with incompatible EPROMs in the rack.

Corrective action: replace the Processor, or replace the Common Memory module if error codes 4.0.- 4.6. remain on.

Run Time Errors

02	Task or Configuration checksum failure

Corrective action: clear the error in the error log. This will also clear the LEDs. Reload the parameter files and application tasks to the UDC module in the slot referenced in the error log.

Runbase Booting

The following status/error codes are displayed while you load the runbase, or operating system, onto the Processor module(s). All of the following codes except 6.5. apply to the top port of the Processor module, labeled “Programmer/Port B”.

5.1.	Incompatible runbase downloaded
6.0.	Unexpected interrupt on upper port of Processor
6.1.	Parity error
6.2.	Receiver overrun
6.3.	Framing error
6.4.	Serial port fatal error
6.5.	Illegal interrupt on lower port of Processor
6.6.	Transmit interrupt error
6.7.	Runbase integrity lost
6.8.	Bad runbase checksum
6.9.	Transmit buffer error
7.0.	Multi-Processor runbase download in progress
7.1.	Disconnect time-out during download
7.2.	Spurious interrupt received

Corrective action: 6.3. may be caused by attempting AutoMax ON LINE PROGRAMMING functions before the runbase is loaded onto the Processor module(s) in the rack. In this case, exit the ON LINE PROGRAMMING menu and download the runbase. 7.0 is a status message only. For all other error codes, cycle power and try to load the runbase again.

Loading the Runbase over the Network

8.0.	Bad message length specified for network message
8.1.	Bad destination drop
8.2.	Transmitting drop inactive
8.3.	Destination port unallocated
8.4.	Destination port busy
8.5.	Did not receive expected response
8.6.	Spurious network interrupt received
8.7.	Network message is being transmitted

Corrective action: 8.0. and 8.1. are caused by a failed Processor in the leftmost slot. For 8.2., check the coax cable; then try replacing the network module. For 8.3. - 8.5., check the destination Network module, then the leftmost Processor in the destination rack. For 8.6. and 8.7., cycle power and try to load the runbase again.

Miscellaneous Process Errors

8.8	Processor failure

Corrective action: replace Processor module.

STOP ALL Error Codes

The following hardware and software error codes cause all tasks running in the rack to stop.

10	Event count underflow - too many WAITs (max. 32768) - not enough SETs (BASIC tasks)
11	Event count overflow - too many SETs (max. 32767) - not enough WAITs (BASIC tasks)
12	Hardware event time-out - interrupt time exceeded programmed time-out limit in a Control Block task

13	Runbase boot error - a check on the runbase failed
14	Processor overlap limit exceeded - ran out of processing capacity (time)
15	External watchdog time-out detected - another Processor in the same rack stopped
17	Address error detected - caused by a read/write to an invalid address
18	Spurious interrupt or hardware failure
19	Power failure detected
1A	Watchdog on this Processor failed
1b	Hardware event count limit exceeded - too many interrupts set without being acknowledged - program too long - collective scans too fast
1C	Illegal instruction detected - runbase software fault - bad processor module - bad EPROMs
1d	Privilege violation detected - runbase software fault - bad processor module
1E	Un-implemented instruction detected - runbase software fault - bad processor module
1F	Illegal interrupt detected - runbase software fault - bad processor module
31	Bus error - attempt to access invalid address
32	Define channel error - problem in application software
33	Define scan error - hardware fault
34	Memory integrity lost - hardware fault
35	DC drive CML block initialization error
36	Communication between drive Processor and I/O controller lost
37	DC drive I/O controller run-time board error - hardware fault
38	UDC module generated a STOP ALL
39	UDC module interrupt allocation failed
3A	Processor OS incompatible with UDC OS

Corrective action: correct the problem in application software. Try to reset by cycling power and re-loading configuration and application tasks. Replace the Processor module. For error code 31, see J-3650; for error code 37, see J-3669. For error code 17: If you define bits in a register that is also defined as a register, neither the bits nor the register can be forced. For error code 38, examine the error logs for all UDC tasks in the rack. Error code 38 can be caused by enabling CCLK on more than one module in the rack. Verify that CCLK is enabled on only one module in the rack. For error code 39, cycle power to the rack and re-load the configuration and application tasks. For error code 3A, check OS compatibility.

BASIC STOP ALL Error Codes

The following error codes are caused by problems in BASIC tasks and cause all tasks to stop.

40	Too many RETURNs from GOSUBs (or RETURN without GOSUB)
41	Illegal jump into a FOR loop
42	NEXT statement does not match current FOR
43	Invalid START EVERY statement
44	Invalid EVENT statement
45	STOP statement executed in application software (causes a STOP ALL/CLEAR)
46	SET or WAIT attempted with no event definition
47	Task stack overflow
48	GOSUBs not balanced at END statement
49	Insufficient space for channel buffer
4A	Attempted to execute undefined opcode
4B	Attempted to execute non-executable opcode
4C	Attempted to execute illegal opcode
4D	RESTORE to non-DATA statement line number
4E	Attempted to take square root of a negative number
4F	Attempted RESUME without being in an ON ERROR handler

Corrective action: correct the problem in application software. Error code 47 can be caused by performing a PUT on a closed port, PRINT statements, GOSUB without RETURN, and deeply-nested subroutines. Error code 4A can be caused by attempting to use Ethernet functions with the standard operating system loaded. To use Ethernet functions, you must load the Ethernet operating system.

Multibus™ and Processor Bus STOP ALL Error Codes

50	Onboard parity error
51-54	Onboard bus error or access violation
55	Multibus parity error during read access
56-58	Multibus access violation or bus error
60	Network interrupt allocation failed
61	Network receiving queue overflow
62	Network transmit queue underflow

Corrective action: reset by cycling power and re-loading configuration and application tasks. If the small green LED labeled “OK” on the Processor module faceplate is off, replace the Processor module. Correct any incorrect accesses in application software. Systematically swap out hardware modules. For error codes 55-58, if none of the above correct the problem, try replacing the backplane.

AutoMax Drive-Related Error Codes

The following error codes indicate a power circuit or external drive system fault. They apply only to systems that incorporate DCS 5000 micro-regulators. After correcting the problem, reset the Processor module by cycling power and re-loading the configuration task and application tasks to clear the error code. Note that these error codes also appear in the Error Log for the Processor.

80 Instantaneous overcurrent fault
- armature current exceeded IOC_THRESH value in CML task

81 Line sync loss fault

82 Tach loss fault
- 40% armature phase angle with less than 5% tach feedback

83 Overspeed/overvoltage fault
- CML task OSV_FDBK exceeded OSV_THRESH number

84 Hardware overspeed fault
- drive analog module potentiometer setting exceeded by input voltage

85 External IET fault
- external fault input triggered

86 Phase rotation fault
- incorrect phasing

87 Shorted SCR detected in power module

Corrective action: troubleshoot power circuit and external drive system. These errors do not cause a Stop All and do not require re-loading the configuration or application tasks.

Configuration Error Codes

The following error codes usually indicate a discrepancy between the actual hardware configuration and the I/O definitions in the configuration for the rack.

E0 TASK specified in configuration uninstalled, at wrong priority, of wrong type, on wrong Processor module; wrong spelling of TASK

E1 Invalid configuration, configuration not successfully downloaded.

E2 I/O referenced in configuration is missing.

E3 I/O referenced in configuration is missing. Invalid configuration, configuration not successfully downloaded.

E4 Error building task, insufficient memory in Processor Module. Invalid configuration, configuration not successfully downloaded.

E5 Error building task, insufficient memory in Processor Module. Invalid configuration, configuration not successfully downloaded.

E6 I/O referenced in configurations is missing. Error building task, insufficient memory in Processor Module.

E7 Invalid configuration, configuration not successfully downloaded. I/O referenced in configuration is missing. Invalid configuration, configuration not successfully downloaded.

E8 Error installing application task, common symbol could not be resolved, insufficient memory in Processor Module.

E9 Error installing application task, common symbol could not be resolved, insufficient memory in Processor Module. Invalid configuration, configuration not successfully downloaded.

EA	Error installing application task, common symbol could not be resolved, insufficient memory in Processor Module. I/O referenced in configuration is missing.
Eb	Error installing application task, common symbol could not be resolved, insufficient memory in Processor Module. I/O referenced in configuration is missing. Invalid configuration, configuration not successfully downloaded.
EC	Error building task; and error installing application task, common symbol could not be resolved, insufficient memory in Processor Module.
Ed	Error building task; and error installing application task, common symbol could not be resolved, insufficient memory in Processor Module. Invalid configuration not successfully downloaded.
EE	Error building task; and error installing application task, common symbol could not be resolved, insufficient memory in Processor Module. I/O referenced in configuration is missing.
EF	Common variable forced by another Processor Module.

Corrective action: verify that the configuration correctly describes the physical configuration of the system and the tasks installed on the Processor module(s). Reset by cycling power and re-loading the configuration and application tasks. For error code EF, un-force the variable and do a STOP ALL from the AutoMax ON LINE PROGRAMMING menu.

Fatal Errors

The following error codes usually indicate that the runbase is not functioning correctly. If any of these error codes appears, the configuration and all application tasks are deleted from the Processor module.

F0-F9	Fatal error
FA-FF	Fatal error

Corrective action: cycle power. Re-load the configuration and all application tasks. Replace the Processor module.

Informational Messages

The following codes signify a particular condition, not necessarily an error.

dd	This Processor module has successfully completed power-up diagnostics and is waiting for other Processor modules to complete their diagnostics
L0	The runbase needs to be loaded onto the rack
b0	Rack configuration is being validated
d0	Application task installation in progress
d1	Waiting on synchronizing event (in a rack with multiple Processors)
d2	Waiting on mutual exclusion lock (in a rack with multiple Processors)

Corrective action for b0 and d0 that do not change or disappear: re-load configuration and application tasks.

Appendix C

AutoMax Task Execution

In AutoMax applications that incorporate multi-tasking, task execution is determined by eligibility to run. A task's eligibility, in turn, is determined by its priority, scan time, and any links to other tasks built into the task. This appendix describes task execution for AutoMax application tasks. Refer to the appropriate Configuration and Programming instruction manual for a description of the task execution for UDC tasks.

Priority

Task priority refers to the relative importance of a task in the application. Tasks whose execution is more critical to the operation of the controlled machinery should be assigned higher priority. Priority can range from 4 (highest) to 11 (lowest). Priority is specified by the programmer when adding a task to the rack.

Scan Time

Scan time refers to how often the task is scheduled to be executed. It is usually measured in ticks. You can assign a tick rate for each AutoMax Processor. The tick rate can range from 0.5 ms to 10.0 ms. The default tick rate is 5.5 ms. See section 6.1 of this manual for more information. For example, an AutoMax task with a 20 tick scan time (using the default tick rate) means that the task is scheduled to start every 110ms.

Scan time is specified differently for PC/Ladder Logic, Control Block, and BASIC tasks. In each case, before setting the scan time, it is important to know approximately how long the task takes to execute. For example, a task that takes 20ms to execute cannot be assigned a scan time of 2 ticks because a 20ms task cannot be started every 11ms. This situation would result in an overlap error (error code 14 would be displayed on the Processor and all tasks in the rack would be stopped).

Scan time for PC/Ladder Logic tasks is set using the PC Editor in the Executive software. The F3 key, followed by I, allows the programmer to enter the scan time. The screen display will indicate the approximate execution time of the task, determined by the Executive software. The AutoMax Ladder Logic language instruction manual contains execution time estimates for each type of Ladder Logic operation.

For AutoMax Control Block tasks, the scan time is specified in ticks using the SCAN_LOOP statement. The approximate execution time of the task can be calculated using the execution time estimates found in the AutoMax Control Block language instruction manual (J-3676).

The scan time for BASIC tasks is usually specified in the START EVERY statement. The START EVERY statement, which is optional, allows the programer to use seconds, minutes, and hours, as well as ticks, as the time unit. Note that there is no simple method of estimating execution time for BASIC tasks ahead of time. One method of estimating execution time for a specific BASIC task is to include a statement that turns on a DC output at the beginning of the task (after the START EVERY statement, if used) and turns it off at the very end of the task. Then the task is put into run with no other tasks running, and an oscilloscope is used to measure the time that the output is on.

Synchronizing Tasks

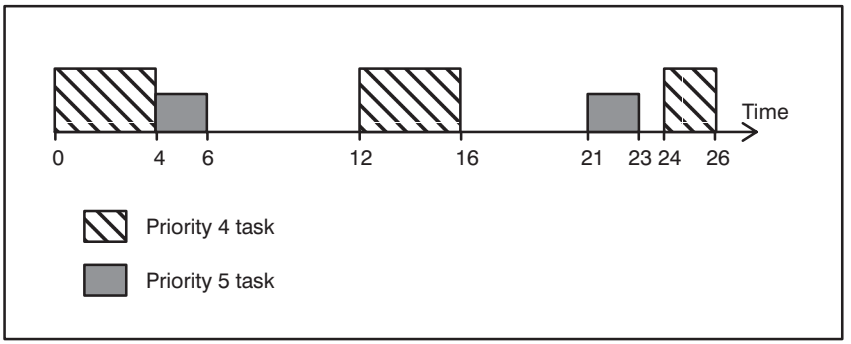
The implications of scan time and priority are particularly important in applications that require tasks to share data or to be synchronized based on hardware or software events. Priority, however, cannot be used to synchronize tasks on different Processors. This is because each Processor controls only the scheduling of the tasks that reside on it. Therefore, the concept of task priority is relevant only for the Processor on which the tasks reside. Task synchronization between AutoMax Processors and UDC modules is handled by interrupts. See the DPS Configuration and Programming instruction manual for more information about UDC tasks.

In addition to priority, there are other methods of synchronizing tasks that work with both tasks on the same Processor and tasks on different Processors. In BASIC tasks, statements such as DELAY, WAIT, SET, OPEN, and INPUT can be used for this purpose. In Control Block tasks, the EVENT parameter in the (required) SCAN_LOOP statement is used to synchronize tasks. PC/Ladder Logic tasks use the EVENT coil for task synchronization.

Examples of Task Execution

The scan time required for the highest priority task on the AutoMax Processor should be used as a guide to determine how tasks should be scheduled. It is also necessary to factor in system overhead, e.g., communication with the personal computer, when assigning priority and scan time. Total Processor utilization should be kept to 80-85%. Processor utilization can be monitored using the Info/Log option in the Executive software.

See the figure below for an example of task execution. Assume you have a priority 4 PC/Ladder task that needs to be executed approximately every 12 ticks and that takes 4 ticks to execute. Assume you also have a less critical priority 5 BASIC task that needs to be executed approximately every 17 ticks and takes 2 ticks to execute. The PC/Ladder task would start at time 0, then finish execution at 4 ticks. At this point the BASIC task would be eligible to run (its time 0).

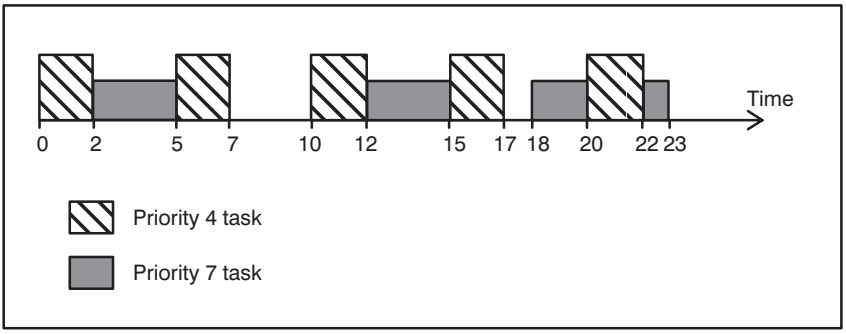


If there are any scheduling conflicts between tasks, i.e., more than one task is ready to run, priority takes precedence over scan time. A higher priority task that is scheduled to run can suspend a lower priority task, regardless of whether the lower priority task is ready to begin execution or has already begun executing. The suspended task is allowed to execute or continue execution from where it was suspended only after the higher priority task is finished.

Consider the following example. There are two tasks on a Processor: a priority 4 Control Block task that needs to be executed every 5 ticks and a priority 7 BASIC task that needs to be executed every 8 ticks. Assume the Control Block task requires 2 ticks to execute and the BASIC task requires 3 ticks to execute.

The Control Block task starts, and finishes executing 2 ticks later. At this point, the BASIC task is eligible to run (its time 0). When it finishes, the time elapsed from the point of view of the Control Block task is 5 ticks and the higher priority task can then execute again. At 10 ticks, however, both the Control Block and the BASIC task are scheduled to run. Because the Control Block task has a higher priority, it will execute first. The BASIC task will execute immediately afterwards.

Note that although the BASIC task was prevented from executing at the originally scheduled 10 ticks, it is still scheduled to run again at the next 8-tick interval, i.e., at 18 ticks. Two ticks into this scan, however, it will be suspended by the Control Block task which is scheduled to run at 20 ticks. After the Control Block task completes its scan, the BASIC task will continue the remainder of its scan.



Initial Scan

In applications that require multiple interdependent tasks, the programmer needs to ensure that the structure of the individual tasks takes into account the possible timing lag between execution of different types of tasks. The first scan of the task is used to execute only the initialization code for the following types of tasks: all BASIC tasks with a START EVERY statement, all PC/Ladder Logic tasks, and all Control Block tasks.

Control Block initialization code consists of all statements before the SCAN_LOOP statement. BASIC task initialization code consists of all statements before the START EVERY statement. PC/ladder Logic task initialization code is constructed by the Executive software.

BASIC tasks without a START EVERY statement do not have initialization code and are always executed in total (if they are not suspended by a higher priority task which is scheduled to run) during their first scan. Execution of BASIC tasks without a START EVERY statement is based only on their priority or their links to other tasks because there is no method of establishing scan time without a START EVERY statement. Note that this type of BASIC task is executed only once unless it includes a GOTO statement that results in a loop.

Appendix D

Limitations on Object File Size

This appendix explains application and configuration object file limits for AutoMax Processor modules. Note that there are limits both on the size of tasks that can be compiled and on the size of tasks that can be loaded onto the rack.

When BASIC, Control Block, and Configuration files are compiled, the object, symbol, and data size are printed to the screen and totaled. For Control Block tasks, estimated execution time is also printed to the screen. Note that for BASIC and Control Block tasks, the data required for arrays is included in the total displayed but does not actually affect the maximum allowable task size.

Task Size Limitations

	Object	Symbol	Data
Configuration file	*	**	**
BASIC task	***	64K	***
AutoMax Control Block task	32K	<-------- 32K -------->	
UDC Control Block task	20K	<-------- 20K- -------->	
	# of Symbols		
PC/Ladder Logic task	2047		

* The configuration file has an object size of 0 unless the reconstructible option is used.

** The size limit for the configuration file is determined by the amount of memory available on the Common Memory module (M/N 57C413B or M/N 57C423), or if a Common Memory module is not being used, the AutoMax Processor that will store the configuration file.

*** The size limit for the object and data is determined by the amount of memory available for application tasks on the AutoMax Processor that will run the task. Refer to the AutoMax Processor Module instruction manual (J-3650) for this information.

Determining the Size of Individual Tasks

For Control Block, BASIC, and Configuration files, after the task is compiled, the pertinent information is printed to the screen both in terms of object, symbol, and data, and in total. To the total figure add approximately 1K for system overhead. This figure will tell you how much memory on the Processor or UDC each task requires. You can also choose to create log files when you compile or verify application tasks. The log file will contain the same statistics printed to the screen.

For PC tasks, the command F3 I in the PC editor is used to determine both the number of symbols and the approximate amount of memory the task will require on the Processor (this includes system overhead).

Application Task Memory Utilization in the Rack

See section 10.0 for the amount of memory available on each Processor module. In racks with a single Processor, add the total size of all the application tasks and the configuration to determine the amount of memory they will require on the Processor. In racks with multiple Processors, the configuration file will reside on the Common Memory module (M/N 57C413 or M/N 57C423 in slot 0). To determine the amount of memory that will be utilized on each Processor, add the total size of all the application tasks that will reside on each Processor.

In rare cases, PC/Ladder Logic tasks that are within the limit specified above may not be loadable on a Processor. This is due to the manner in which Processor memory is allocated for bit variables in PC tasks. The smallest amount of Processor memory that can be allocated for any PC task variable is one word (16 bits). Each 16-bit register used in the PC task requires one word of memory. Individual bits within registers also require one word of memory. However, if more than one bit in a register is used, all those bits can be stored in one word of memory.

When there is a Common Memory module located in slot 0 of the rack (i.e., being used for bus arbitration and common memory storage) there are approximately 1400 words (2.8K) of memory available on each Processor for storing PC task bit variables. If there is no Common Memory module in the rack, or the Common Memory module is in a slot other than 0 (i.e., serving as extra memory), there are approximately 1000 (2K) words of memory available on each Processor for storing PC task bit variables. The M/N 57C435 Processor has approximately 2000 words (4K) of memory available for storing PC task bit variables regardless of whether or not a Common Memory module is in the rack.

This method of allocating space for bit variables will not cause a problem unless you use a large number of single bits within different words. Note that if you have run out of room for PC tasks for this reason, you can still load BASIC and Control Block tasks up to the limit of the Processor's memory.

Note that each physical online connection (maximum of four, one to each Processor) made through the Executive software either over the network or directly to the rack will require an additional 5K of memory on the Common Memory module. If there is no Common memory module, the 5K required will be allocated from the Processor.

Appendix E

Windows Command Summary

(in chart form)

Access via:	Keyboard	Mouse
Control menu	- ALT,SPACEBAR	click on menu
Menus	- ALT,RIGHT(or LEFT),ENTER - ALT,underlined letter	click on menu
Commands	- DOWN(or UP),ENTER - type underlined letter	click on command
Making a Selection	- 1. DOWN (or UP) to locate item 2. SPACEBAR to select	click on item
Making Multiple Selections	- 1. CTRL+DOWN(or UP) to locate item 2. CTRL+SPACEBAR to select 3. Repeat for each item to be selected	CTRL+ click on each item to be selected
Moving in a Dialog Box	- 1. TAB to area, 2. Use DIRECTION keys - ALT+underlined letter	click on desired area, click on desired item
	- SPACEBAR toggles choices	click to toggle
Text Box	- 1. Use DIRECTION keys to locate text 2. SHIFT+DIRECTION to select text	drag to select text
List Box	- 1. Use DIRECTION keys, HOME, END, PAGE UP, PAGE DOWN 2. SPACEBAR to select	use scroll bar, then click on item
Moving a Window or Icon	- 1. ALT+ESC 2. ALT+SPACEBAR 3. press M 4. use DIRECTION keys 5. ENTER	window - drag title bar Icon - drag icon
Changing Size of a Window	- 1. ALT+ESC 2. ALT+SPACEBAR 3. press S 4. use DIRECTION keys to select and move border 5. ENTER	drag border or corner

Access via:		Keyboard	Mouse
Enlarging a Window	-	1. ALT+ESC 2. ALT+SPACEBAR 3. press X	click on Maximize box (upper-right corner)
Shrinking a Window	-	1. ALT+ESC 2. ALT+SPACEBAR 3. press N	click on Minimize box (upper-right corner)
Restoring a Window	-	1. ALT+ESC 2. ALT+SPACEBAR 3. press R	click on Restore box (upper-right corner)
Restoring an Icon	-	ALT+TAB	double-click on icon
Scrolling:			
rows,columns	-	Use DIRECTION keys	drag scroll box
Top of list	-	HOME	drag scroll box to top of scroll bar
End of list	-	END	drag scroll box to bottom of scroll bar
up 1 screen	-	PAGE UP	click above scroll box
down 1 screen	-	PAGE DOWN	click below scroll box
Cancel a Command	-	ESC	click on empty space
End session	-	1. ALT+first letter of left-most menu 2. X to select Exit Windows command. 3. OK at dialog box	double click on Control menu, then click on OK at dialog box.

Appendix F

Database Definitions

The relevant fields in each of the AutoMax Executive V3.6 databases are described here. Any fields not identified here are reserved for future use and should be left unused. All of the following database files are intended as read only.

Database: $SYSTEM.DBF

FIELD	TYPE	DESCRIPTION
ID	CHARACTER	The name of the system
DESC	CHARACTER	The description of this system
ENGINEER	CHARACTER	The user responsible for this system
CUSTOMER	CHARACTER	The end user of this system
SYSTEM_VER	CHARACTER	1 for V3.3 and later, otherwise 0.

Database: $SECT.DBF

FIELD	TYPE	DESCRIPTION
ID	CHARACTER	The name of the section
DESC	CHARACTER	The description of this section
ENGINEER	CHARACTER	The user responsible for this section

Database: $RACK.DBF

FIELD	TYPE	DESCRIPTION
ID	CHARACTER	The name of the rack
SECT_ID	CHARACTER	The name of the section containing this rack
RACK_PART	NUMERIC	The rack part code from the rack permanent table
LOCATION	CHARACTER	The physical location of this rack
DESC	CHARACTER	The description of this rack
ENGINEER	CHARACTER	The user responsible for this rack

Database: $IORACK.DBF

FIELD	TYPE	DESCRIPTION
MAST_SLOT	NUMERIC	Slot in the master rack which contains the remote I/O card to which this rack is connected (00-15)
REM_DROP	NUMERIC	This rack's drop on the remote I/O network (1-7)
PORT	NUMERIC	This rack's port on remote head, if applicable (0-3)

LOC_PORT	NUMERIC	This rack's port on local head, if applicable (0-3)
LEVEL	NUMERIC	The code identifying this rack's location in the remote I/O topology (see the diagram below)
RACK_PART	NUMERIC	The rack part code from the rack permanent table
LOCATION	CHARACTER	The physical location of this rack
DESC	CHARACTER	The description of this rack
ENGINEER	CHARACTER	The user responsible for this rack

Database: $CARD.DBF

FIELD	TYPE	DESCRIPTION
MAST_SLOT	NUMERIC	The slot in the master rack which contains this card or which contains the remote I/O card which is the remote I/O network master (00-15)
REM_DROP	NUMERIC	If this card is in a remote rack, this is the drop number for that remote rack (1-7); otherwise, it is 0
PORT	NUMERIC	Port on remote head, if applicable (0-3)
LOC_PORT	NUMERIC	Port on local head, if applicable (0-3)
LEVEL	NUMERIC	The code identifying this card's location in the remote I/O topology (see the diagram below)
SLOT	NUMERIC	The slot number containing this card
CARD_PART	NUMERIC	The card part code from the card permanent table
NETWORK	CHARACTER	Network identifier; for network cards only (A-Z)
NET_DEPTH	NUMERIC	Network depth; for network cards only (1-55)
NET_DROP	NUMERIC	Drop number; for network and gateway cards only
DESC	CHARACTER	The description of this card
ENGINEER	CHARACTER	The user responsible for this card
WIRE_NOTE1	NUMERIC	For AutoMax Processors (M/N 57C430A, 57C431, and 57C435), the high order byte of the AutoMax Processor tick rate (in microseconds) For the UDC module (B/M 57552 and B/M 57652), the drive type for drive A (see Parameter Record Types table for values)

		For the Shark Scanner module (M/N 57C554), the value is one (1) for an all-digital rack or zero (0) for a mixed rack.
WIRE_NOTE2	NUMERIC	For AutoMax Processors (M/N 57C430A, 57C431, and 57C435), the low order byte of the AutoMax Processor tick rate (in microseconds) For the UDC module (B/M 57552 and B/M 57652), the drive type for drive B (see Parameter Record Types table for values)

Note: *Memory variables are stored in slot 99.*

Database: $SWVAR.DBF and $NET.DBF

FIELD	TYPE	DESCRIPTION
MAST_SLOT	NUMERIC	The slot in the master rack which contains this card or which contains the remote I/O card which is the remote I/O network master (00-15)
REM_DROP	NUMERIC	If this card is in a remote rack, this is the drop number for that remote rack (1-7); otherwise, it is 0
SLOT	NUMERIC	The slot in which the variable resides (00-15)
REGISTER	NUMERIC	The multibus register number (0-65535)
BIT	NUMERIC	The bit number within the multibus register for boolean variables (0-15)
LVAR_NAME	CHARACTER	The variable without type specifier or array dimensions; for network variables, this is the local variable name
GVAR_NAME	CHARACTER	The variable without type specifier or array dimensions; used for the network-wide name for network variables only
NETWORK	CHARACTER	Network identifier; used for network variables only (A-Z)
DESC	CHARACTER	The description of this variable
ARRAY_DIM1	NUMERIC	First dimension; used for memory variables only
ARRAY_DIM2	NUMERIC	Second dimension; used for memory variables only
ARRAY_DIM3	NUMERIC	Third dimension; used for memory variables only
ARRAY_DIM4	NUMERIC	Fourth dimension; used for memory variables
DATA_TYPE	NUMERIC	The type of this variable: 8 Floating point 16 ! Double integer

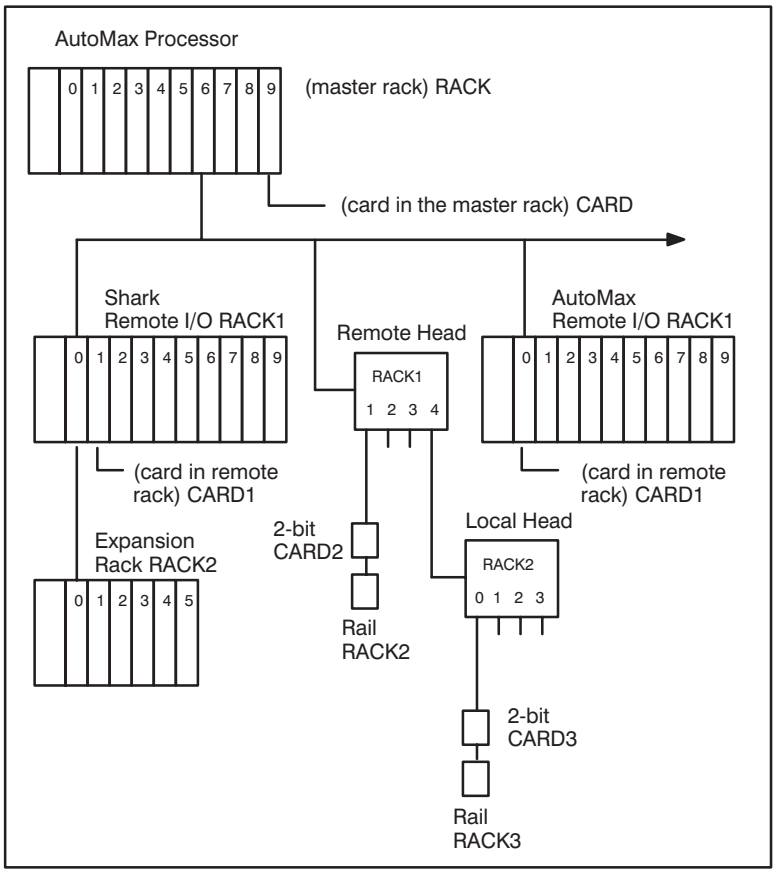
32	%	Integer
64	@	Boolean
128	$	String

CARD_TYPE NUMERIC	The content of this field depends on the data type. For string variables, it contains the string length (0-255). Zero (0) represents the default string length of 32 characters. This field is not used for card type.
DEF_TYPE NUMERIC	This field is used to designate whether a COMMON memory variable is volatile (2) or non-volatile (3). For all other variables, this field will be zero (0).

Database: $TASKS.DBF

FIELD	TYPE	DESCRIPTION
NAME	CHARACTER	The task name without extension, drive, or directory; unique within this rack
TYPE	CHARACTER	The task type: PC, BLK, BAS, or INC
SLOT	NUMERIC	The slot number of the Processor in which this task will be loaded (0-4, 0-15 for UDC tasks))
PRIORITY	NUMERIC	The priority of this task (4-11)
DESC	CHARACTER	The description of this task
ENGINEER	CHARACTER	The user responsible for this task
CRITICAL	BOOLEAN	T if the task is critical
RECONSTR	BOOLEAN	T if the task should be compiled so that it can be reconstructed from object code
UTILITY	BOOLEAN	T if the task is a utility task

LEVEL CODES



Level	Code stored in databases
SYSTEM	1
SECTION	2
RACK	3
CARD	4
RACK1	5
CARD1	6
RACK2	7
CARD2	8
RACK3	9
CARD3	10

The following two database definitions detail the fields in the permanent tables. These databases are intended for read-only use. They are primarily used to translate the part type codes stored in other databases to Reliance part numbers.

Database: $PTCARD.DBF

FIELD	TYPE	DESCRIPTION
PART_TYPE	NUMERIC	Card part code
PART_NUM	CHARACTER	Reliance part number
FAMILY	NUMERIC	Family code
GROUP	NUMERIC	Group code
ABBR	CHARACTER	Abbreviation for the card
DESC	CHARACTER	Description
BITMAP_ID	NUMERIC	Bitmap ID code

Database: $PTCARD.DBF

FIELD	TYPE	DESCRIPTION
PART_TYPE	NUMERIC	Rack part code
PART_NUM	CHARACTER	Reliance part number
DESC	CHARACTER	Description

Database: $PARAM.DBF

FIELD	TYPE	DESCRIPTION
MAST_SLOT	NUMERIC	Master rack slot (0-15)
REM_DROP	NUMERIC	RIO drop number (0-7)
PORT	NUMERIC	Remote port (0-3)
LOC_PORT	NUMERIC	Local port (0-3)
LEVEL	NUMERIC	Rack level code (see table below)
SLOT	NUMERIC	Target card slot (0-15)
SUB_ID	CHARACTER	Sub ID (A-Z) needed when there is more than 1 parameter record per card
TYPE	NUMERIC	Parameter record type (see table below)
VERSION	NUMERIC	Format version for the given type (0-255)
P01-P50	NUMERIC	Unsigned word parameters 1-50 (0-65535) (see table below)

Rack Level Codes

LEVEL	VALUE	DESCRIPTION
RACK	3	For cards in the master AutoMax rack
RACK1	5	For cards in the AutoMax remote I/O rack and PMI rack
RACK2	7	For cards in Shark expansion racks
RACK3	9	For cards in I/O rails

Parameter Record Types

PARAMETER RECORD TYPE	VALUE
UDC unused	1
UDC SD3000 6-pulse	2
UDC SD3000 12-pulse	3
UDC SD3000 12-pulse auxiliary	4
UDC SF3000 field regulator	5
UDC SA3000 vector	6
UDC SA500 vector	7
UDC SA500 brushless	8
UDC VZ3000 Vector	9
UDC VZ3000 Vector	10
UDC VZ3000	11
UDC VZ3000	12
UDC SB3000	13
UDC SA3000 Constant Power	14
UDC SA3000 Volts/Hertz	17
UDC SA3100 Constant Power	22
UDC SA3100 Volts/Hertz	23

Appendix G

Importing Racks from Systems Containing Network (.NET) Files

If your system has .NET files, you must run NETIN prior to importing AutoMax V2 racks. Use the following procedure to compile the .NET files and generate the network-wide database. When this procedure is complete, you can import AutoMax V2 racks.

Step 1. Create the system by using the System Add procedure described in section 5.1.1. The name must be the same as the source AutoMax V2 system name.

Step 2. Access the Control menu (ALT,SPACEBAR or click on the Control menu).

Step 3. Use Switch To to access the Program Manager.

Step 4. Access the Resource group and select NETIN, or select Run from the File menu, enter NETIN.EXE on the command line, then select OK. The Network Import dialog box will be displayed.

Step 5. Enter information for the following fields, then select OK.

Source Drive: Enter the drive containing the AutoMax V2 system.

Destination Drive: Enter the drive containing the destination V3 software.

Destination Library: Enter the destination V3 library name.

System Name: Enter the AutoMax V2 system name.

Step 6. Use Switch To to return to the current Executive software.

Note: *This procedure will create .NET and .OBN files in the current Executive system directory. After the racks have been imported, these files can be deleted.*

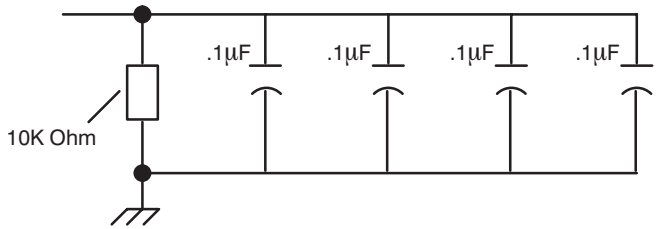
Appendix H

Using Foreign Modules in an AutoMax Rack

A foreign module must meet the following hardware and software requirements before it can be used in a local or remote AutoMax rack. See the AutoMax Power Supply Module and Racks Instruction Manual (J2-3008) for more information.

Hardware Requirements

1. The foreign module must conform to general IEEE 796 specifications. The AutoMax rack will support foreign modules which meet the following level of IEEE Standard bus compliance: D16M20I16.
2. For foreign modules that have field connections, the field connections must be electrically isolated because the rack does not have a direct connection to ground. See the figure below for more information.



3. The BCLK (Bus Clock) signal is activated only by the presence of the Common Memory module (M/N 57C413 or 57C423). Consequently, if the foreign module requires this signal, the Common Memory module must be in slot 0 in the rack.
4. For foreign modules that can serve as bus masters, i.e., can control the bus, if the rack already contains a Processor, the Common Memory module (M/N 57C413 or 57C423) must be in slot 0 in the rack.
5. The bus arbitration method used by AutoMax systems is parallel round-robin. Any foreign module that can serve as a bus master must support parallel round-robin bus arbitration. Bus arbitration is supported only in slots 1 through 4. Bus masters can be placed only in these slots.
6. The CCLK (Constant Clock) signal is activated only through one of the following modules: Resolver Input module (57C411), 2-Channel Analog Input module (57C409), UDC module (57552), or the Pulsetach module (57C421). If the foreign module requires the CCLK signal, one of these modules must be in the rack and the command to activate CCLK using the appropriate bit in one of these modules must be issued in an application task.
7. Foreign modules must not generate interrupts.
8. For foreign modules in remote racks, inputs must respond only to Multibus memory read. Outputs must respond to both Multibus memory read and write. If this is not the case, the foreign module will be treated as an output module only by the remote I/O system.

9. Each slot has 64K of address space. Foreign modules that have more memory can be addressed in the rack if the proper number of slots to the right of the foreign module are left empty. For example, a foreign module with 128K of memory can be used fully by leaving the slot to the right of the foreign module empty (64K + 64K = 128K total).
10. Foreign I/O modules that drive pins 24 and 26 on the bus cannot be used in the rack.
11. All modules in local racks must respond to requests on the bus within 10 μsec. All modules in remote racks must respond to requests on the bus within 3 μsec.

Software Requirements

For use in local racks, the foreign module must support 20-bit address lines.

For use in remote racks:

1. The foreign module must support 20-bit address lines;
2. The foreign module must support 16-bit data transfer only;
3. The foreign module must be memory-mapped, not I/O mapped;
4. The foreign module address must begin on a 64K boundry.

Accessing Foreign Modules

If a foreign module meets all the hardware requirements and software requirements above, it can be used in an AutoMax rack. For local racks, the method of accessing the foreign module depends upon further software specifications of the foreign module. There is only one method of accessing a foreign module in a remote rack.

Accessing a Foreign Module in a Local Rack

Method A

Requirements:

1. The module must be capable of 16-bit data transfers.
2. If it is an output module, it must be able to respond to read as well as write commands from the AutoMax Processor.
3. Addresses must begin on a 64K boundry.
4. Input and output registers must be a separate addresses.

Reference to the appropriate variable names in application tasks.

Method B

No requirements beyond the Hardware Requirements and Software Requirements listed above.

Access the module through BASIC IOWRITE and IOREAD statements in application tasks.

Accessing a Foreign Module in a Remote Rack

No requirements beyond the Hardware Requirements and Software Requirements listed above.

Reference to the appropriate variable names in application tasks.

Appendix J

Quickstart Guide

WARNING

ONLY QUALIFIED PERSONNEL FAMILIAR WITH THE CONSTRUCTION AND OPERATION OF THE CONTROLLED EQUIPMENT SHOULD INSTALL, ADJUST, OPERATE, AND/OR SERVICE THIS EQUIPMENT. READ AND UNDERSTAND THIS MANUAL AND OTHER MANUALS APPLICABLE TO YOUR INSTALLATION IN THEIR ENTIRETY. FAILURE TO OBSERVE THIS PRECAUTION COULD RESULT IN BODILY INJURY OR DAMAGE TO EQUIPMENT.

This appendix is intended for individuals who want to start using the AutoMax Programming Executive as quickly as possible. It will be particularly useful for those who are familiar with the Windows software and with previous versions of the Programming Executive. Familiarity with basic concepts in the instruction manual is assumed. This appendix presents the typical sequence of steps followed when using the Programming Executive, but includes little of the detail or explanatory material found in the instruction manual. This appendix also does not describe any online functions, e.g., monitoring.

Note that this appendix does not describe how to import application tasks created with earlier version AutoMax or DCS 5000 Programming Executives or with the version 8.x AutoMax Software Developer (ASD).

Help screens are available for most commands. Use the F1 key to click Help to access help for any screen where F1 is displayed.

Installing Windows and the AutoMax Programming Executive

1. If you have not already done so, install Windows V3.1 software according to the directions in the Windows instruction manual.
2. If you have not already done so, install a mouse. Note that although most operations will be easier to use with a mouse, a mouse is not required.
3. Install the Programming Executive using the instructions found in the Installation manual, J2-3107. Instruction for installing the separately-purchased DPS software options can be found in the same manual.
4. Type AUTOMAX3 to run both Windows and the AutoMax Programming Executive software. The first screen displayed after the name and copyright information is the System Configurator, which can be thought of as the “main menu,” since all other options are chosen from this screen. Choose the Setup menu and then select one of the following commands.

   Select AutoMax to run the AutoMax configuration procedure. This will allow you to make changes to the AutoMax Executive configuration (see section 5.5.2).

   Select Communication to make changes to the communication port and baud rate used for online communication (see section 5.5.3).

   Select Printer to set the parameters for printer output (see section 5.5.4).

Configuring the Application

Assuming you have already selected all the hardware required for your application and have determined its physical configuration, you can begin describing this configuration with the Executive software. The description of the hardware in your application, as well as information about the application tasks you create for that application, will be stored in databases which are transparent to the user. Application tasks themselves are stored in conventional files.

5. Create systems for your application. From the pull down menu labeled Systems in the System Configurator, select the Add command to “add” the system to the database. You will be prompted for information like the system name, engineer, etc. Enter the information in the dialog box and click on OK when complete. Select Document from the same pull down menu to create/edit a text file that contains information about the system.
6. Create sections for each system. The system for which you will create sections must be selected (highlighted on the screen). Select the Add command from the pull down menu labeled Sections to “add” the section to the database. You will be prompted for information like the section name, engineer, etc. Enter the information in the dialog box and click on OK when complete. Select Document from the same pull down menu to create/edit a text file that contains information about the section.
7. Create racks for each section. The section for which you will create racks must be selected. Select the Add command from the pull down menu labeled Racks to “add” the rack to the database. Fill in the appropriate information in the dialog box, e.g., the number of slots in the rack, etc., and click on OK when complete. Select Document from the same pull down menu to create/edit a text file that contains information about the rack.

Configuring Modules in the Rack and Configuring I/O

When you have finished creating systems, sections, and racks, you can configure the modules in the racks. The rack for which you will configure modules must be selected (highlighted) first. The configuration procedure results in a graphic representation of the rack and the modules in the rack.

8. Select the Configure Rack command from the Racks pull down menu. Select a slot in the rack by moving the mouse to the desired slot and clicking.
9. Select the Add command from the Configure menu. Scroll through the list of modules displayed until the desired module is highlighted. Click on OK in the dialog box. Selecting certain modules may result in prompts for other information. Refer to the instruction manual for the module in question concerning specific configuration information, as well as this instruction manual. Note that UDC module configuration is complex and will require consulting the appropriate configuration and programming instruction manual.

If you add Remote I/O modules, you can zoom in (using the Zoom In command from the GoTo pull down menu) to configure the remote I/O section (remote racks or Remote I/O Heads) of the installation. If you select Heads and then add a Rail, you can double-click on the Rail to configure the I/O modules on the Rail. Use Zoom Out from the GoTo menu to zoom out to the local rack.

If you add a UDC module, you can zoom in (by double-clicking on the UDC module or by using the Zoom In command from the GoTo pull down menu) to configure the drive hardware. From the drive hardware configuration screen, you can zoom in to add digital rails and two-point modules. You can access the Parameter Entry screens by selecting the Configure Parameters command from the Configure menu or by zooming in on the appropriate drive hardware.

If you make a mistake, select the Remove command from the Configure menu to remove the selected module.

Configuring Variables

Once all of the hardware for a rack has been configured, you can configure the variables in each rack. Configuring variables consists of attaching symbolic names to the variables that must be accessible to more than one application task in the rack. There are two categories of variables that are configured: I/O locations that are connected to field signals and the common memory variables on Processor modules and Common Memory modules. Local variables, i.e., those that do not need to be accessed by more than one application task, are not configured. They are defined in the application task in which they are used.

Three different “forms” are used to configure variables: a basic form used for I/O and interface modules (including UDC modules), a form used for common memory modules, and a form for Network Interface modules (M/N 57C404). The correct form, including any pre-defined dedicated registers, is automatically called up when you select a module to configure.

10. To configure variables on a selected module, select Configure Variables from the pull down menu. You can also zoom in to configure the variables by double clicking on the selected module. For Processor and Common Memory modules, the form displays item numbers to be configured, not register or bit numbers. For modules with I/O, registers and bits that need to be configured will be displayed on the screen, in pages and views where applicable. Views are logical collections of registers, e.g., interrupt registers. A Page is a contiguous set of 64 registers within a view. If multiple views exist, a dialog box will prompt you for the view you want to configure.

The Programming Executive software incorporates three features that allow you to configure variables more quickly: copy/paste, auto-name, and substitute. All three of these options are found on the Configure Variables pull-down menu.

Creating Application Tasks and Adding Them to the Rack

After configuring the variables in the rack, you can create application tasks for the rack. Before you can create a new task, you must first add the task to the rack.

11. Select the Manage Tasks command from the Rack menu. Select the Add command from the Task menu. Select New on the resulting dialog box for a new task. Enter the task name, along with the other information prompted for in the next dialog box. This procedure adds the name of the application task and all information about it to the selected rack. The application task can now be created/edited.

12. Select the Edit Task command from the Task menu (or double-click the selected task). The Programming Executive will call up the appropriate editor (PC editor for PC tasks or text editor for all other types). See the appropriate language instruction manuals for more information. See this instruction manual for information on the PC editor.

13. When you are finished editing the task, save it and select Compile from the Manage Tasks menu. .PC and .INC tasks do not need to be compiled. Compilation statistics (or errors) will be printed to the screen and to log files if the log option is selected.

Generating a Configuration File and (Optional) Parameter Object File

Before you can load application tasks to the AutoMax Processors and UDC modules in the rack, you must generate a configuration file for the rack and parameter object file(s) for the UDC module(s). These files will be downloaded to the rack first. The rack for which the configuration will be created must be selected first.

14. Select the Generate Configuration command from the Command pull down menu in the Task Manager.
15. Exit the Task Manager.

Loading the Operating System onto the Rack

Before loading any application tasks onto the rack, you must connect the cable between the personal computer and the leftmost Processor module in the rack and load the operating system onto the rack. **Loading the operating system onto the AutoMax Processors or UDC modules in the rack deletes any existing operating system and all application tasks in the rack.**

16. Select the Command pull-down menu from the System Configurator. Select Load Operating System to load the operating system to all AutoMax Processors and/or UDC modules in the rack, and optionally, to all Processors in all racks on the network.

Loading the Configuration File, (Optional) Parameter Object File, and Application Tasks

17. Select the Manage Tasks command from the Rack menu at the System Configurator. Select ON_LINE from the Task Manager screen. At this point you can load the configuration file, the parameter object file(s), and all application tasks for the selected rack. The configuration file, always called _CONF.CNF, must be loaded first or at the same time as the application tasks.

Refer to this instruction manual for information about online operations.

Appendix K

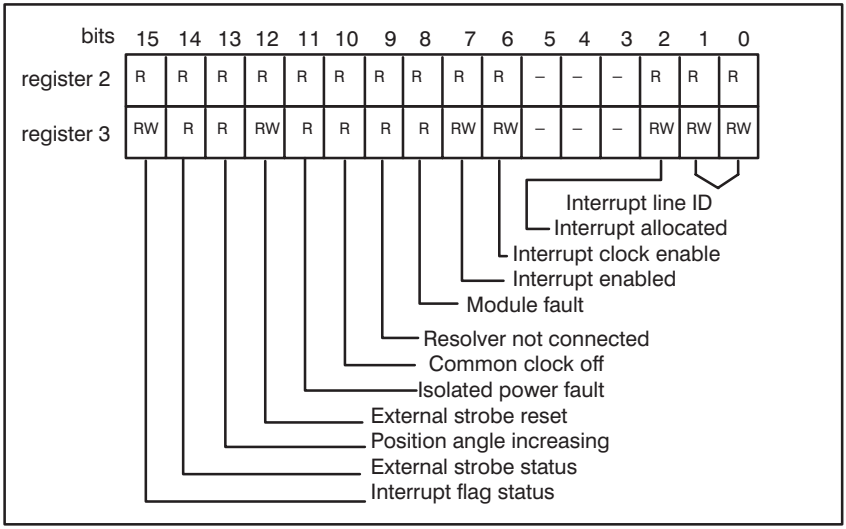
Import Notes

The procedure for importing racks and tasks from AutoMax Executive V1 or V2 is described in detail in section 5.3.9. This appendix describes some anomalies that may occur in the import procedure. See Appendix Q for information about importing racks and task from earlier V3 Executives.

For some modules, registers defined in AutoMax V2 cannot be configured in the current Executive software. Before you import a system or rack, first check (and edit, if necessary) your AutoMax V2 configuration task. Otherwise, an error message may be displayed (invalid/incompatible data type) when you try to configure the module. See the examples that follow.

Resolver Input Module (M/N 57C411) Example

Registers 2 and 3 of the Resolver Input module are illustrated below:



Registers 2 and 3 each define the same variables. Register 2 is read only ,"R," and contains only "R" bits. Register 3 is read/write, "R/W," and it contains both "R/W" and "R" bits. The current version of the Executive software allows data to be read only from an "R" register. Therefore, all of the above variables can be read from register 2 only. Data can be written to register 3 only.

Before executing the Import procedure, examine your AutoMax V2 configuration task to see how you are using registers 2 and 3. Make any necessary changes to ensure that you are reading from register 2 only and writing to register 3.

DCS Drive Controller (M/N 57406) Example

The registers listed below are used to configure the DCS Drive Controller.

AutoMax V2		Current Version Executive AutoMax		
Register	**Bit**	**Register**	**Bit**	**Function**
1		4096		Analog Reference Input
2		4097		Analog Tach Feedback
3		4098		Armature Voltage Feedback
4		4099		Major Loop Current FDBK
5		4100		AC Line Voltage FDBK
41		4136		Tach Loss Delta Threshold
42		4137		PLL Max. Phase Change
43		4138		PLL Max. Period Change
51	0	4146	0	Fast Bridge Change Enable

Note that the register numbers used in the current version correspond to the old register numbers plus 4095. Before executing the Import procedure, edit your AutoMax V2 configuration task to set the register numbers for this module to their new values by adding 4095 to each register number used for this module.

Appendix L

AutoMax Bill of Material Generator

After you have completed the physical configuration of a system, you can generate a bill of material which lists all of the racks, heads, rails, and modules used in the system along with the needed batteries, cables, software, and optional hardware. This bill of material can be used to get a preliminary idea of the hardware and software needed for your system. However, you should consult with your Reliance representative or distributor for assistance in configuring your system as cost-effectively as possible.

The bill of material is divided into three sections (see the figure below). The first section lists the number of the racks, heads, rails, and modules in the specified system. The second section lists the batteries and cables for racks or modules that typically require them. For example, the Network Communications module (M/N 57C404) requires a Communications Passive Tap (M/N 57C380) and a cable (M/N 57C381) to connect the Network Communications module to the Passive Tap. One Passive Tap and one cable are included for each Network Communications module. This section also lists the optional hardware and software that can be used with the specified system. Extra items, such as network tee connectors, are listed without a quantity. The last section lists all of the generic cards used in the system along with the descriptions (part number, functional description, etc.) assigned to them when they were configured.

System: 135792 Page: 1
Date: 03/19/93
Customer:
Engineer:bagwell

Quant	Part Number	Description
1	45C1	Digital Input/Output Rail
6	45C40	Dual 120VAC/DC, Input
2	45C60	Dual 120VAC Output (2 Amps)
1	57C330	Remote I/O Head
1	57C332	AutoMax 10 Card Slot Rack
1	57C404	Network Interface
1	57C416	Remote I/O Interface
1	57C430	AutoMax 6010 Processor
1	57C552	Universal Drive Controller Interface
1	60000	Power Module Processor
1	60001	Power Module resolver feedback
1	60002	Power Module Technology – DC Drive
1	61C346	Analog Rail, 4 Channel 0–10VDC Input
1	GEN32	Generic card with 32 registers
1	GEN32K	Generic card with 32,768 registers
1	UDCPMI	AutoMax UDC PMI 3 Card Slot Rack

System: 135792 Page: 2
Date: 03/19/93
Customer:
Engineer:bagwell

Quant	Part Number	Description
1	45C70	Coax BNC Tee Connector
	45C71	Coax BNC 75 Ohm Terminating Load
	45C72	Coax BNC Male Connector
2	57C380	Communications Passive Tap
2	57C381	Cable – Comm. module to passive tap
	57C395	AutoMax Programming Executive
	57C493	
	61C126	ReSource Portable Computer

System: 135792 Page: 3
Date: 03/19/93
Customer:
Engineer:bagwell

Rack	Slot	Description
A00	1	Generic card with 32,768 registers
A00	7	Generic card with 32 registers

Use the following procedure to generate a bill of material for an AutoMax system. The Microsoft Windows User's Guide provides information about using the Program Manager and the File Manager.

Step 1. From your current AutoMax application, access the Control menu (ALT,SPACEBAR or click on the Control menu).

Step 2. Use Switch To to access the Program Manager.

Step 3. Access the Resource Group and select GEN B/M. The AutoMax Bill of Material dialog box will be displayed.

Step 4. Enter any information needed for the following fields. Use TAB or SHIFT + TAB to move to the next field. The entries for drives and library directory must match your Executive software configuration as entered in Setup.

System Name: Enter the name of the system.

Library drive: If different than the default, enter your library drive.

Library directory: If different than the default, enter your library directory.

Drive for temporary files: If different than the default, enter the drive used for temporary workspace.

Step 5. After entering the last field, press TAB or ENTER to begin generating the bill of material file. When the file is complete, the Resource Group of the Program Manager will be displayed.

The bill of material file is contained in the system directory (<d>:\<library>\<system>) and is named <system>.BM.

Step 6. To print the bill of material for a system, access the Main Group and select the File Manager.

Step 7. Access the system directory that contains the desired <system>.BM file.

Step 8. Choose <system>.BM from the list of files, then select Print from the File menu.

Step 9. When the bill-of-material file has finished printing, use the Control menu to close the File Manager. Then minimize the Program Manager to return to your application.

Note: *The library directory contains the source file (GENBM.PRG) for the AutoMax Bill of Material Generator. This file is intended to be a read-only file. You can access it by using your text editor.*

Appendix M

Copying Systems and Racks

The procedures that follow describe how to copy an existing AutoMax system or AutoMax rack. These procedures use various DOS commands. Refer to your DOS manual if you need additional information about DOS commands.

Copying a Rack

You can copy an AutoMax rack within a selected system, from one system to another within the same AutoMax library, or to a system in another AutoMax library. The new rack will be an exact copy of the source rack, including all of the tasks for the rack.

Note**:** *If network-wide names are used in a rack which is being copied from another system, the network-wide names will not be found in the network database. Before copying the rack, you should either unlink the local and network-wide names or add the network names to the network database.*

Start at the System Configurator screen. Use the following procedure to copy a rack.

Step 1. Use the Add command from the Rack menu to add the new rack to the selected system and section.

Step 2. Write down the directory paths for the rack you are copying and for the new rack, i.e., <DRV>:\<LIB>\<SYSTEM>\<SOURCE_RACK> and <DRV>:\<LIB>\<SYSTEM>\<NEW_RACK>.

Step 3. Exit the Programming Executive and return to DOS.

Step 4. Use the DOS **CHDIR** command to go to the <NEW_RACK> directory.

Step 5. Use the DOS **DELETE** command to delete all the files in the <NEW_RACK> directory.

Step 6. Use the DOS **COPY** command to copy all the files in the <SOURCE_RACK> directory to the <NEW_RACK> directory.

Step 7. Return to the Programming Executive. The new rack will be an exact copy of the source rack.

Copying a System

You can copy an AutoMax system within the same AutoMax library or to another AutoMax library. The new system will be an exact copy of the source system, including all of the racks and tasks for each rack.

Start at the System Configurator screen. Use the following procedure to copy an entire system.

Step 1. Use the Add command from the System menu to add the new system.

Step 2. Write down the directory paths for the system you are copying and for the new system, i.e., <DRV>:\<LIB>\<SOURCE_SYS> and <DRV>:\<LIB>\<NEW_SYS>.

Step 3. Exit the Programming Executive and return to DOS.

Step 4. Use the DOS **CHDIR** command to go to the <NEW_SYS> directory.

Step 5. Use the DOS **COPY** command to copy the $SYSTEM.DBF and $SYSTEM1.NDX files to another directory.

Step 6. Use the DOS **DELETE** command to delete all the files in the <NEW_SYS> directory.

Step 7. Use the DOS **XCOPY** command with the /s switch to copy the subdirectories and files from the <SOURCE_SYS> directory to the <NEW_SYS> directory.

Step 8. Use the DOS **COPY** command to copy the previously saved $SYSTEM.DBF and $SYSTEM1.NDX files back to the <NEW_SYS> directory.

Step 9. Return to the Programming Executive software. The new system will be an exact copy of the source system.

Appendix N

Tokens and Locking

Tokens provide a means of write-protecting the databases and tasks for one or more racks or an entire system when using the AutoMax Programming Executive in a network environment. Locking will prevent more than one user at a time from changing the same database information.

Tokens

When more than one copy of a file exists, there is the potential that the various copies will be modified independently. This problem can occur in AutoMax when part or all of a system is transferred to another workstation. To avoid losing independent changes, only one copy can be considered the “source” copy. All other copies must be considered to be read-only. Tokens are used in AutoMax to ensure that only one copy can be modified.

A token is an abstract entity which represents the ability to modify a group of files. There will be one token to represent the system database group and the system and section documentation files, and a separate token for each rack and its associated files. If a copy of the rack file “holds” the token, that copy can be modified. If a copy does not hold the token, it cannot be modified. Only one copy of the system files or of a rack file can hold the token. If a user attempts to make modifications without the token, the user will receive an error message like the following:

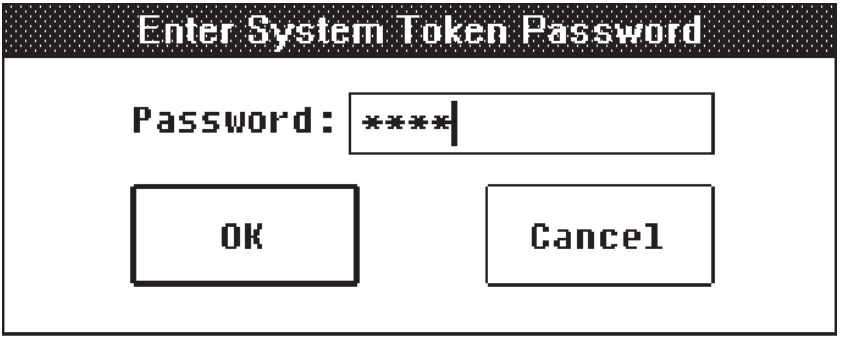
When a system or rack is created, a token is created along with it. Whenever the system (or rack) is transferred using the Transfer command in AutoMax, the user has the option of transferring the token with it. If the token is transferred out, the original copy becomes read-only and only the destination copy can be modified.

Token Passwords

When an AutoMax system is located on a personal computer which is used by more than one person, or on a network server, unauthorized changes to the system files must be prevented. This is accomplished through the use of token passwords. In other words, in addition to having the token for the file in question, the user must also provide a password before making changes.

As part of the AutoMax Programming Executive installation procedure, the system administrator can decide whether a single system-wide password or individual passwords for each rack within a system will be used. The password(s) can be set by using the Set Password command. The password can be from 0 to 15 characters. If no password is entered, the password will default to <null>.

If a token's password is <null>, the user will not be asked for the password before being allowed to make changes. This makes the password transparent for the user who doesn't need it. If the password is not <null>, the following dialog box will appear when the user attempts to make a change to any of the files protected by the token:



As the password is being entered, the characters will echo to the screen as asterisks (*).

Once entered, up to 16 passwords are stored for the duration of the current Windows session. These will be valid whenever the password is required again. The user will be asked for the password only when none of the stored passwords match. If the user works in more than 16 racks with different passwords, only the last 16 passwords entered will be stored by AutoMax. If a password no longer on the list of 16 is required, the user will have to enter it again before modifying the rack.

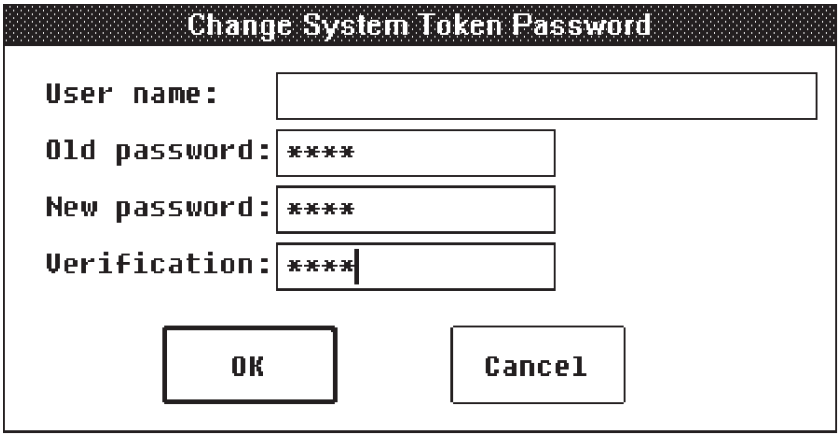
If a single system-wide password is used, the user will have to enter the password only once when the first modification is made.

Token Passwords in Transfer and VCL

When a system or rack is transferred to another workstation, or placed in the Version Control Library, the tokens can go with it. If tokens are transferred, you have the option of transferring the token password or clearing the password. If the token password is transferred, the user on the receiving end will be required to enter the token password in order to access the system or rack.

Changing Token Passwords

The **Set Token Password** command on the System and Rack menus can be used to change the token password for the system or a selected rack. When the command is executed, the following dialog box will be displayed:



The following fields must be entered:

User Name: A user name of up to 30 characters. The default is the name entered in the AutoMax Setup.

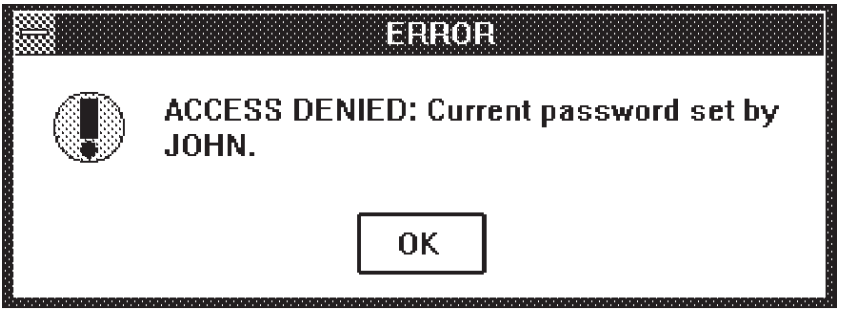
Old Password: The old password. Each character will be echoed as an asterisk (*). This information is required to verify that the user is authorized to change the password. If the old password was <null>, this field will not appear and any user may change the password.

New Password: A password of up to 15 characters. Each character will be echoed as an asterisk (*).

Verification: The new password again. Each character will be echoed as an asterisk (*). This field is required to make sure the user typed the new password correctly. If this entry does not match the new password, the screen will display an error message and the cursor will be returned to the New Password field.

Token Password User IDs

When a token password is changed, the user will be asked to enter his User Name. The default will be the name entered in the AutoMax Setup. When a user enters the wrong password, a message such as the following will be displayed:

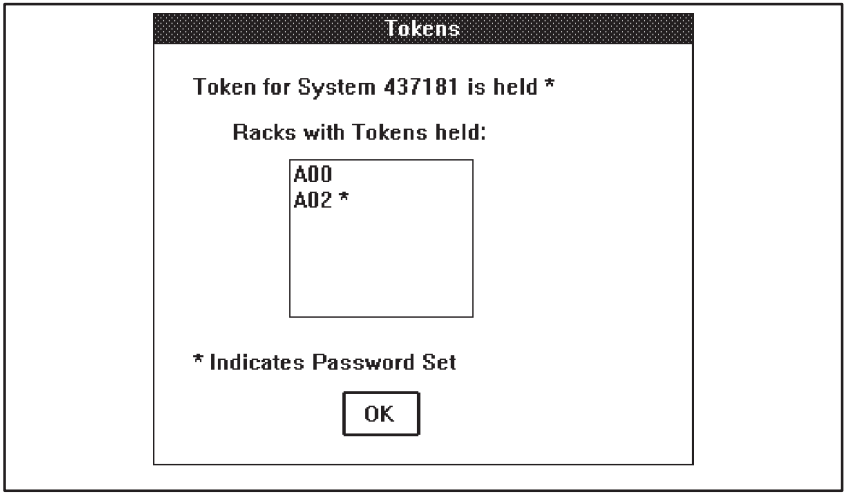


This assumes that the person who set the password wanted to be identified and entered his name when setting the password. When a token is transferred (or placed in the Version Control Library) along with its file, both the User ID and the password are deleted from the file.

The User ID will be stored at offset 0 in the token file (rack_name.TOK) so that external utilities (not part of the AutoMax Programming Executive) can be used to extract the names and generate reports of which users are working on which systems or racks in a system.

Show Tokens Command

The **Show Tokens** command appears on the System menu in the System Configurator. This menu item can be selected only when there is a single system selected. When this command is selected, a dialog box will be displayed stating whether the system token is held for the currently selected system. It will also have a list box showing all of the racks in the current system for which the token is held. “Held” means that the system or rack information is modifiable. An asterisk (*) next to the system or rack name indicates that there is a non-null password set for that file’s token. See the example below:



The Token Manager Utility

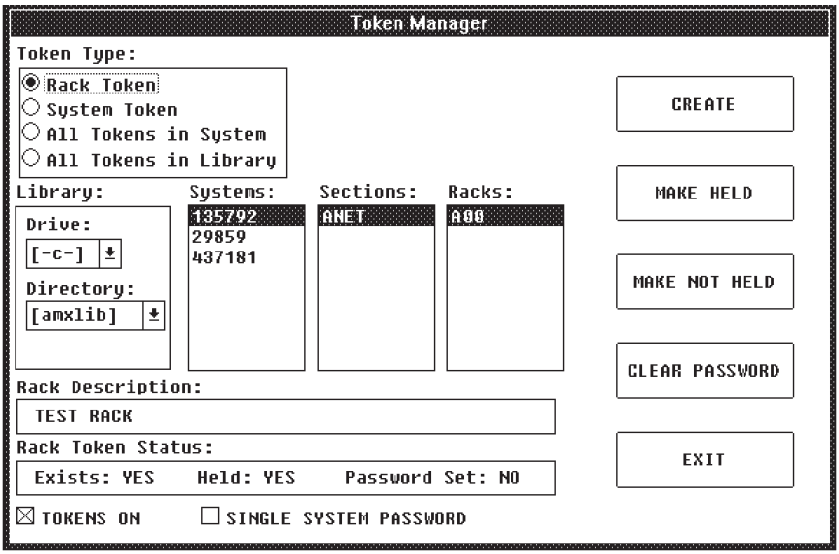
The Token Manager utility (TOK_MAN.EXE) is used to manage tokens for AutoMax systems. This utility can be used to create any system or rack token (held or not held) and let you clear the token password for any system or rack token. The Token Manager is provided on the distribution disks, and is installed with the rest of the AutoMax Programming Executive software.

Use the following procedure access the Token Manager.

Step 1. From your current AutoMax application, access the Control menu (ALT,SPACEBAR or click on the Control menu).

Step 2. Use **Switch To** to access the Program Manager.

Step 3. Access the ReSource Group and select TOKEN MANAGER. The Token Manager dialog box will be displayed. See the figure below.



Step 4. Select the drive and AutoMax library directory. All of the systems contained in the selected library will be listed.

Step 5. Select a system and section. All the racks that belong to that section will be displayed. Database description information will be displayed at the bottom of the dialog box. Note that you can skip this step if you are going to create all the tokens for the selected library.

Step 6. Select the type of token you want to work on. The choices are described in detail below.

Rack Token

If a rack is selected:

The token status line will display information about the rack token for the selected rack.

If the rack token does not exist, the commands MAKE HELD, MAKE NOT HELD, and CLEAR PASSWORD will be disabled.

If the rack token does exist and is held, the commands CREATE and MAKE HELD will be disabled. If the token password has not been set, CLEAR PASSWORD will also be disabled.

If the rack token exists but is not held, the commands CREATE, MAKE NOT HELD, and CLEAR PASSWORD will be disabled and the information about whether the password is set will not be displayed.

If no rack is selected, all the commands except EXIT will be disabled and the token status information will not be displayed.

System Token

If a system is selected:

The token status line will display information about the system token for the selected system.

If the system token does not exist, the commands MAKE HELD, MAKE NOT HELD, and CLEAR PASSWORD will be disabled. Information about whether the token is held and whether the password is set will not be displayed.

If the system token does exist and is held, the commands CREATE and MAKE HELD will be disabled. If the token password has not been set, CLEAR PASSWORD will also be disabled.

If the system token exists but is not held, the commands CREATE, MAKE NOT HELD, and CLEAR PASSWORD will be disabled and information about whether the password is set will not be displayed.

If no system is selected, all the commands except EXIT will be disabled and the token status information will not be displayed.

All Tokens in System

If a system is selected:

The token status information will not be displayed.

If the system token does not exist, the commands MAKE HELD, MAKE NOT HELD, and CLEAR PASSWORD will be disabled.

If the system token does exist and is held, the command CREATE will be disabled.

If the system token exists but is not held, the command CREATE will be disabled.

If one of the commands MAKE HELD, MAKE NOT HELD, or CLEAR PASSWORD is selected, and not all the rack tokens for the system exist, a message will be displayed for each non-existing

rack token, asking whether the user would like to create it. Processing will continue following each response.

If no system is selected, all the commands except EXIT will be disabled and the token status information will not be displayed.

All Tokens in Library

If this option is selected, all the commands except CREATE and EXIT will be disabled. System, section and rack information will not be displayed and the token status information will not be displayed.

Step 9. Select one of the following commands:

CREATE - Creates “held” token(s).

MAKE HELD - Makes existing token(s) “held” regardless of current state.

MAKE NOT HELD - Makes existing token(s) “not held” regardless of current state and clears the token password.

CLEAR PASSWORD - Clears the token password on selected token(s).

EXIT - Exits the Token Manager utility.

Hierarchical Locking

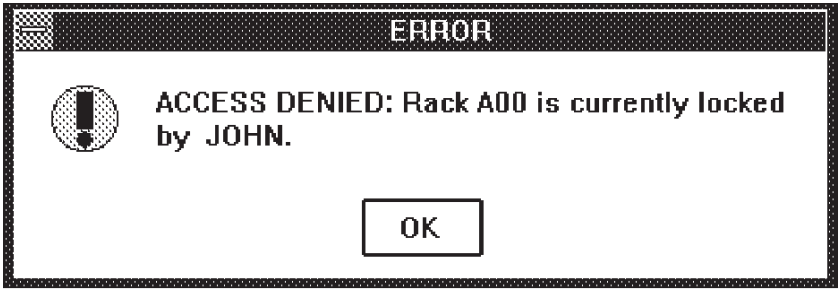
When a user makes modifications within a system, the AutoMax system will lock the appropriate files to ensure that the user has exclusive access to them. Once these files are locked, any other user who attempts to modify them will see a message on his screen indicating that the files are locked by another user. When the modifications to the files are complete and the databases have been updated, the lock will be released. The files will then be accessible for modification by another user.

Note that only the files that are accessed through the Programming Executive will be locked. Files that are called up from within the text editor will not be locked, and will not respect locks that may have been taken on them.

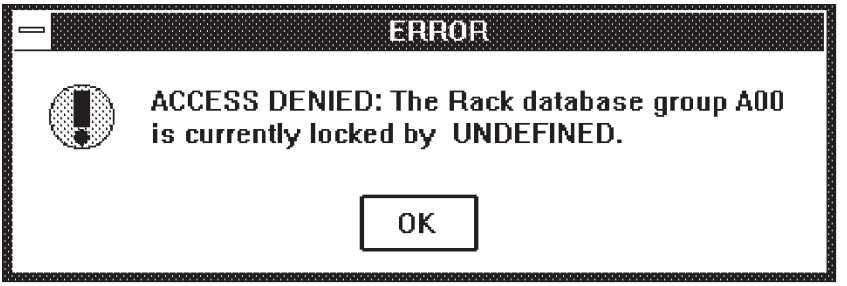
A lock can be taken not only for files, but for whole racks, sections, or an entire system. Higher level locks protect all the files under them. For example, holding a section lock prevents other users from modifying any of the files that pertain to that section (racks or tasks). Lower level locks also provide file protection. For example, if a rack is locked, the section or system to which it belongs cannot be deleted.

Setting the User Name

When a lock is taken, the AutoMax system stores the username (entered in the AutoMax Setup - up to 30 characters) with the lock. This name is used in error messages to other users who attempt to modify a locked file. The error message displayed when someone attempts to access a locked file would look something like the following:



If the User Name was undefined when the lock was taken, the message would look something like the following:



When another user is working at a lower level than the level you are trying to lock (e.g., editing a task in the rack you are trying to transfer), the AutoMax system must search for the lock file in order to identify the user who is blocking your lock. If the offending lock is removed before it is found, it may not be possible to identify the user. In this case, your lock would still fail and the message would look something like the following:

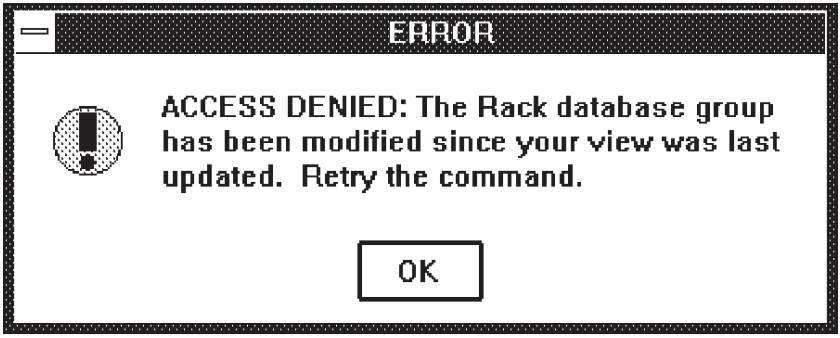
Database Lock Time/Date Stamps

In order to create display screens for the user, various AutoMax applications read information from the databases and store the information in memory. An example is the rack diagram displayed by the Rack Configurator. In a network environment, users A and B both might read RACK2 into memory. User B modifies the information and writes his changes to the database (which releases the lock) before user A attempts to also modify the rack. User A may then make modifications based on old information. The result could undo user B's changes or result in erroneous information. As long as user B was finished making his changes and had written his changes to the database (released the lock) before user A attempted to take the lock, the locking scheme alone would not prevent problems.

For another example, suppose that user A has selected the Rack Configurator and is viewing a 16-slot rack. User B then modifies the rack from a 16-slot rack to a 10-slot rack using the Modify command on the Rack menu in the System Configurator. User A then proceeds to add a module in slot 15, which is perfectly valid according to the information he has in memory. The result is a module assigned to a slot which no longer exists in the rack.

To avoid this problem, each lock has a time/date stamp that is updated every time the lock is issued. Whenever data is read from the database, the time/date stamp of the lock that protects that database will be saved. When changes are made to the source copy, the lock covering the files will be taken. If the time and date match the time/date stamp saved when the data was read, then no one else has taken this lock and the files have not been changed. If the time or date on the lock is more recent than the time/date stamp that was saved when the files were

read, a lock has been taken in the meanwhile and the files have potentially been changed. In this case, the source file data will be updated, giving the user a correct view of the data as it exists in the databases. The following message will be displayed:



The user will have to re-initiate his command. Since this check is done as soon as the user shows intent to modify, no changes to the database files will be lost.

Lock and Token Handling in Transfer

When transferring into a library that is not on floppy disks, the lock for the appropriate level (System, Section, Rack, etc.) will be taken in the destination library. When multiple tasks are transferred, the lock is taken for the entire rack.

The user will be given the option of transferring any held tokens within the scope of the transfer. If there is a token password, and the user has not entered it previously, the user will be prompted for the password before the transfer will execute. If the user chooses to transfer the tokens, the destination copy will hold the tokens when the transfer is complete. Note that tokens are not required for transferring in either direction. However, if the user tries to transfer on top of the copy that holds the token, he will receive a warning message. The user will be given the option of continuing the transfer or skipping the rack (for racks). If the files being written over have a token password, the user must enter the password before the files protected by the token will be written over. If the copy with the token is written over, the resulting copy will still hold the token. When a token is transferred, the token password can be either transferred or nulled for the destination copy.

Note that transferring racks is not considered overwriting the system token because each rack's database files are protected by that rack's token, not the system token. If the system token is held by the destination copy, and racks are transferred into it, the user will not receive a message asking whether he wants to overwrite the system token. However, the user will receive a warning message if the destination holds the rack token(s).

Lock and Token Handling in System or Rack Import

System and rack import disregard locks. The import procedure will fail if the system or rack being imported already exists in the target library. The import procedure will create tokens for any systems and racks that are imported.

Lock and Token Handling in VCL

When fetching a system or rack(s) from the VCL, the user will be given the option of transferring any tokens held by the Version Control Library (VCL) within the scope of the fetch. If the user chooses to transfer the tokens, the destination copy will hold the tokens when the fetch is complete. Tokens are not required to fetch. However, if the user tries to fetch on top of the copy that holds the system or rack token, he will receive a warning message. The user will be given the option of continuing the fetch or canceling. If the files being written over have a token password, the user must enter the password before the files protected by the token will be written over. If the copy with the token is written over, the resulting copy will still hold the token. When a token is transferred, the password is nulled for the destination copy unless the destination originally held the token and had a password set.

Note that fetching a rack is not considered overwriting the system token because each rack's files are protected by that rack's token, not the system token. If the system token is held by the destination copy, and racks are fetched into it, the user will not receive a message asking whether he wants to overwrite the system token. However, the user will receive a warning message if the destination holds the rack token(s).

When fetching from the VCL, locks are taken out in the AutoMax library for the appropriate system, section(s), rack(s), and task(s). For example, if a user is fetching a section with two racks, a lock will be taken out in the AutoMax library for the section. No one else will be able to modify either of the two racks or the section until the fetch is complete. If a user has selected the Fetch Latest option for the fetch, a lock is taken out in the VCL so that no one can snapshot anything during the fetch.

When snapshotting, the VCL cannot already hold the token of any rack or system group being snapshot. The user will have the option of transferring any tokens held within the scope of the snapshot. If the token is transferred to the VCL, the token password is nulled.

When snapshotting to the VCL, locks are taken out in the AutoMax library for the appropriate system, section(s), rack(s), and task(s). For example, if a user is snapshotting a system with two racks that belong to the same section, locks will be taken out in the AutoMax library for the system, section, and the two racks. No one else will be able to modify the system, section or either of the two racks until the snapshot is complete. Locks are also taken out in the VCL so that two users cannot snapshot the same system, section(s), rack(s), or task(s) at the same time.

Appendix P

Version Control Library (VCL)

The Version Control Library (VCL) provides an interface to the Polytron Version Control System™ (PVCS) by Intersolv™. PVCS is not required to use AutoMax and is not distributed as part of the AutoMax Programming Executive. However, you must have PVCS Version 5.0 (or later) installed on your network (or local) drive in order to use the VCL features in AutoMax Executive V3.4 and later.

If you have specified in AutoMax Setup that you have PVCS installed, **Version Control Library** will appear as a menu item on the System, Section, and Rack menus in the System Configurator, and on the Tasks menu in the Task Manager. When this menu item is selected, a sub-menu listing two items, **Fetch** and **Snapshot**, will be displayed. These two commands are described in detail in the sections that follow.

As you develop your system, you may find it useful to snapshot all or part of the system, either as milestones are reached, or when files are handed off to another user. You can use VCL to save versions of systems, sections, racks, or tasks. You can view the contents of the Version Control Library and fetch previous versions (or the most current version) of the files contained in the VCL. You can retrieve anything that has been previously stored in the library.

Version Control Library Setup

You must have PVCS installed on your network (or local) drive in order to use the Version Control Library feature in AutoMax V3 and later. In addition, the directory where PVCS is installed must appear in the PATH statement in your AUTOEXEC.BAT file. VCL can then be enabled as part of the AutoMax Setup procedure.

Step 1. Select **AutoMax** from the Setup menu.

Step 2. Enter the letter of the drive and the name of the VCL directory in the box labeled “Version Control Library”.

Step 3. Select the PVCS Installed box. If this box is not selected, the Version Control Library selections on the AutoMax menus will be dimmed (inactive).

Note that a PVCS configuration file (VCS.CFG) will be generated by AutoMax and will be added to the directory that contains the AutoMax V3 software (default = AMX3). This file is used by PVCS to find configuration parameters for its commands. If the PVCS configuration file is not in this directory, Fetch and Snapshot operations will return errors.

Saving Files to the Version Control Library

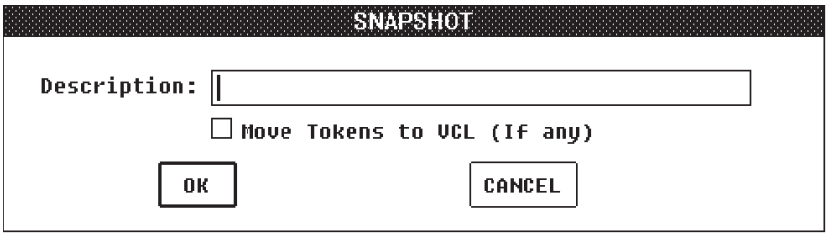
The **Snapshot** command allows the user to save current versions of system files to the Version Control Library. Snapshot can be performed at the following levels: system, section, rack, or task. It may be used with a single item selected or multiple items selected. The items saved to the VCL in a multiple snapshot can be fetched separately.

The same procedure is used to snapshot systems, sections, racks, and tasks. In the following procedure, “SSRT” will represent “system, section, rack, or task.” Note that the double quotation mark (”) is not permitted in the description field for any SSRT.

Step 1. Select one or more SSRT.

Step 2. Select **Version Control Library** from the System, Section, or Rack menus in the System Configurator, or from the Tasks menu in the Task Manager.

Step 3. Select Snapshot from the sub-menu. The following dialog box will be displayed:



Step 4. (Optional) Enter a description of the snapshot (up to 40 characters). The following four characters are not permitted: colon, asterisk, backslash, and dash.

If multiple items have been selected, there will still be only one dialog box and, therefore, only one description for the snapshot. You will, however, be able to fetch each item in a multiple snapshot separately. We suggest that a multiple item snapshot be done only if the single description can pertain to all of the items being snapshot.

Step 5. Select the Move Tokens to VCL check box if you want to transfer any tokens held by the SSRT to the VCL. Note that the check box will not appear if Tokens is not enabled, or if you are snapshotting at the task level.

AutoMax will not allow you to snapshot over tokens that may already be in the VCL. If you attempt to snapshot over tokens, the screen will display an error message and the snapshot will be aborted. See Appendix N for more information about tokens.

Step 6. Select OK to begin the snapshot or Cancel to return to the System Configurator (or the Task Manager).

If you selected OK, a dialog box will be displayed that shows what SSRT is being processed. If you are transferring the SSRT token to the VCL, and the token has a password, you will prompted for the token password before the SSRT will be processed. When processing is complete, a diagram showing the percentage of the snapshot that has been completed will be displayed. When the snapshot is complete, you will return to Windows at the same point as you were before the snapshot (System Configurator or Task Manager). If you are doing a multiple snapshot, the next snapshot operation will begin.

Error Handling during a Snapshot Operation

If an error occurs during the processing of a snapshot operation, an error message will be displayed and the snapshot operation will be aborted. If an error occurs while executing one of the DOS files spawned by the Snapshot command, the user will be returned to Windows and an error message will be displayed. If PVCS fails, the error message will state that PVCS failed and will direct the user to the file PVCS.OUT, which is contained in the AutoMax workspace subdirectory (typically AMXWORK). This file will contain a message describing why PVCS failed. If PVCS did not fail, the user will receive an error message stating only that the snapshot operation failed. The AutoMax library will remain intact, but the VCL may contain only a part of the complete snapshot. If the token was being transferred, it will remain with the source copy of the files unless the snapshot is completed successfully. After determining the cause of the failure, you should execute the snapshot again to ensure that the VCL contains a complete and accurate snapshot.

Tokens and Locking during a Snapshot Operation

If the Move Tokens to VCL check box is selected (checked), then all the tokens in the AutoMax library within the scope of the snapshot will be transferred to the Version Control Library. If the token for the system, section, or rack has a password, the password will be prompted for before the token is transferred to the VCL. However, the user may not write over any tokens already in the Version Control Library. For example, suppose that a user is snapshotting a system with three racks and he has tokens for two of the racks, but the token for the third rack is in the VCL. When he tries to snapshot the system, he will receive an error message telling him that he is trying to write over a system that has its token in the Version Control Library. The snapshot will be aborted before anything is snapshot. The user will need to fetch the third rack from the VCL with the token before he will be allowed to snapshot the system. If the token were held by another user and not the VCL, the snapshot would be successful.

Locks are taken out in the Version Control Library for any system, section(s), rack(s), and task(s) (SSRT) that are being snapshot. For example, if a user is snapshotting a system with two racks that belong to the same section, a lock will be taken out in the Version Control Library for the system, section, and each rack. No one else can snapshot the system, section, or either rack until the first user's snapshot is complete. A lock is also taken out in the AutoMax library so that the SSRT cannot be modified during the snapshot.

Retrieving Files from the Version Control Library

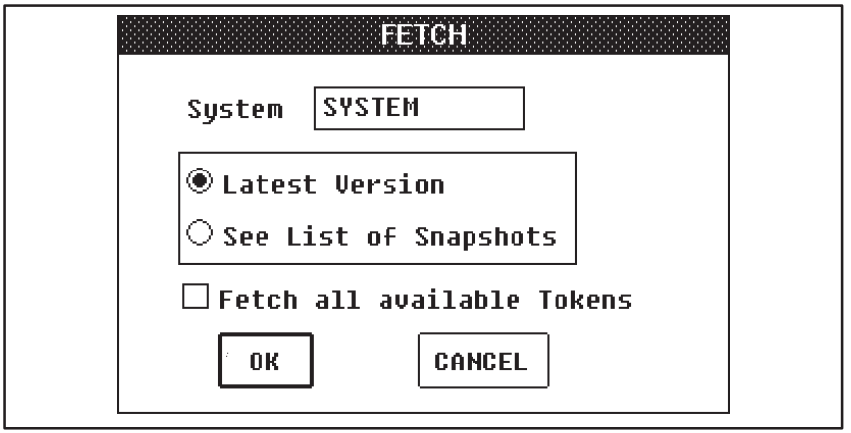
The Fetch command allows the user to retrieve versions of the system files from the Version Control Library. Fetch can be performed at any level to retrieve a single system, section, rack, or task. Fetch can only be used with no SSRT selected or one SSRT selected. In addition, the user cannot fetch the SSRT unless there is exactly one SSRT selected in the levels above. For example, if two sections are multi-selected, or if no sections are selected, Fetch will be disabled under the Rack menu. This rule applies at all levels except the system level.

The same procedure is used to fetch systems, sections, racks, and tasks. Read the remainder of this section carefully before executing the Fetch command. System, rack, and task database files in the AutoMax library can be modified or deleted by the Fetch command.

Step 1. Select **Version Control Library** from the System, Section, or Rack menus in the System Configurator, or from the Tasks menu in the Task Manager.

Step 2. Select Fetch from the sub-menu. The Fetch dialog box will be displayed. If an SSRT was selected, its name will be displayed in the dialog box. If

no SSRT was selected, enter the name of the SSRT you want to fetch. If you are fetching at the task level, you must enter the task name with its extension. See the figure below.



Step 3. The dialog box contains two option buttons. If you want to fetch the latest version of the SSRT, select the “Latest Version” button. If you want to see a list of the previous snapshots from which you can choose the version of SSRT, select the “See List of Snapshots” button.

Step 4. The dialog box has a check box for fetching tokens from the Version Control Library. Select this box if you want to fetch all the tokens held by the VCL (within the scope of the fetch) to the AutoMax library. Note that the check box will not appear if tokens are not enabled or if you are fetching at the task level.

Step 5. Select OK to begin the fetch operation or Cancel to return to the System Configurator (or the Task Manager).

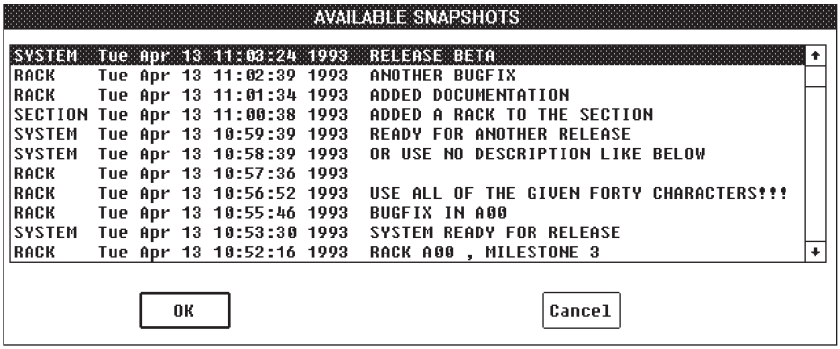
If you selected OK, a dialog box may be displayed that indicates that certain racks are being prepared. This means that the Auto-

Max directories for the racks within the scope of the fetch operation are being emptied of all their AutoMax files. If the directory contains non-AutoMax files, they will remain in the directory. If the directory is empty, it will be deleted. The fetch operation will re-build the AutoMax rack directories with the files contained in the version of the SSRT being fetched. For example, if an AutoMax system contains a section with three racks, and you fetch a version of that section that contains only two racks, the third rack and all its associated task files will be gone.

When empty directories have been deleted, a dialog box will be displayed that shows what SSRT is being processed. When processing is complete, a diagram will be displayed that shows the percentage of the fetch operation that has been completed. When the retrieval of files is complete, you will return to Windows, where a dialog box labeled Rebuilding Databases will be displayed. After the AutoMax databases are rebuilt, a “FETCH COMPLETE” message will appear. You will then return to the same place you were before the fetch operation (System Configurator or Task Manager).

Using the See List of Snapshots Option

If **See List of Snapshots** is selected, the Available Snapshots list box will be displayed before the fetch operation is begun. A sample list box appears below.



The list box will display an entry for every complete snapshot containing the SSRT the user is fetching. For example, if the user is fetching a rack, the list will contain all the rack snapshots of that rack. It will also contain all the section snapshots of the section to which the rack belonged at the time of the snapshot and all the system snapshots of the system to which the rack belonged at the time of the snapshot. For the system and section snapshots, the rack must have belonged to them at the time of the snapshot in order for the snapshot to be on the list. Each reference to a snapshot will contain the level of the snapshot, a time/date stamp, and the description given when the snapshot was done. The references will be in reverse chronological order, where the most recent snapshot is at the top of the list and is selected as the default. If the user selects the most recent (top) snapshot from the list, he will receive a warning that this SSRT may not be the latest version of that entity. He will be given the chance to abort the fetch.

To fetch a particular snapshot, the user needs only to select the one he wants and then select OK. The fetch operation will retrieve the SSRT exactly as it was when it was snapshot. Note that the See List option should only be used when you want to fetch an SSRT in order to restore it to its state at the time of the snapshot.

Using the Latest Version Option

The Fetch Latest Version option allows the user to fetch entities (sections, racks, tasks) from different snapshots to get the latest version of each component entity. In order for entities to be included in a fetch of a higher entity, the entity must have been snapshot at one time and all the entities above it must have been snapshot and belong to the latest version of the entity the user is fetching. The examples that follow illustrate the way in which Fetch Latest Version functions:

In a development environment, the Fetch Latest option could be used as illustrated in the following example. Assume that a section that has 10 racks needs to be developed, and 10 different developers will each be working on his own rack. A supervisor is overseeing the development.

The supervisor could do a snapshot of the section when it contain no racks. Then, as each developer finishes some phase of his rack, the developer could snapshot his rack from the rack level. The supervisor, by fetching the latest version of the section, would receive the latest version of all the racks. When the project was ready for release, the supervisor could fetch the latest version of the section and then snapshot it with an appropriate milestone description.

Example 1:

On Monday, section NET_A was snapshot from the section level at a time when it had only one rack, A00. On Tuesday, another rack, A01, was added to section NET_A and then was snapshot from the rack level.

If section NET_A is fetched from the section level using the Latest Version option, section NET_A will contain both racks A00 and A01.

If section NET_A is fetched from the section level using the See List of Snapshots option, and the snapshot from Monday is chosen, section NET_A will contain only rack A00. This is because the See List of Snapshots option restores the fetched entity to its exact state at the time the snapshot was taken. Rack A01 was not part of section NET_A on Monday, so it is not included in the fetch.

Example 2:

On Monday, section NET_A was snapshot from the section level at a time when it contained no racks. On Tuesday, a rack was

added to NET_A and two tasks were added to the new rack. Each task was snapshot from the task level.

If section NET_A is fetched from the section level using the Latest Version option, section NET_A will contain no racks and no tasks because the rack was never snapshot.

If section NET_A is fetched from the section level using the See List of Snapshots option, and the snapshot from Monday is chosen, section NET_A will contain no racks and no tasks (just as it was at the time of the snapshot).

Example 3:

On Monday, section NET_A was snapshot from the section level at a time when it had only one rack, A00, which contained only one task, TASK1.BAS. On Tuesday, another rack, A01, was added to section NET_A, containing two tasks, TASK2.BAS and TASK3.BAS. TASK2.BAS and TASK3.BAS were then snapshot from the task level.

If section NET_A is fetched from the section level using the Latest Version option, section NET_A will contain only rack A00 because rack A01 was never snapshot. TASK1.BAS, belonging to rack A00, will be fetched. TASK2.BAS and TASK3.BAS, belonging to rack A01, will not be fetched. Remember, in order for an entity to be included in a fetch using the Latest Version option, all the entities above it must also have been snapshot and must belong to the latest version of the entity being fetched.

Error Handling during a Fetch Operation

If an error occurs during the processing of a fetch operation, an error message will be displayed and the fetch operation will be aborted. If the error occurs while in DOS, the user will be returned to Windows and one of two error messages will be displayed. If PVCS failed, the message will state that PVCS failed and will direct the user to the file PVCS.OUT, which is contained in the AutoMax workspace directory (typically AMXWORK). A message in this file will describe why PVCS failed. If PVCS did not fail, the user will receive an error message that says only that the fetch failed. The Version Control Library will remain intact, but the AutoMax library may contain only a part of the entity being fetched. If tokens were being transferred, they will remain with the source copy of the files unless the fetch operation is completed successfully. After determining the cause of the failure, you should execute the fetch operation again to clean up the AutoMax library.

Tokens and Locking during a Fetch Operation

If the check box labeled Fetch all available Tokens is selected (checked), then all the tokens in the Version Control Library within the scope of the fetch operation will be transferred to the AutoMax library. If the user wants to write over any files whose tokens are already in the AutoMax library, he may be required to enter the token password.

When using both the See List of Snapshots option and the option to fetch all available tokens, be careful when selecting a snapshot from the list box. There is a chance that modifications that were made to the files in the VCL will be lost. For example:

> John and Tim both have a copy of the same system and are using the same Version Control Library. They are both working on system SYS1, which has one section, SECT1, and two racks, RACK1 and RACK2. John has the token for RACK1 and has been making changes to RACK1. Tim has the token for RACK2 and has been making changes to RACK2.
>
> John finishes his changes to RACK1 and does a section snapshot of SECT1. He uses the "Move Tokens to VCL" option because he is done modifying RACK1. (John probably should have only done a rack snapshot since he held the token for RACK1 only.) After the snapshot is complete, the token for RACK1 is held by the Version Control Library.
>
> Tim finishes his changes to RACK2 and does a rack snapshot using the "Move Tokens to VCL" option. He uses the "Move Tokens to VCL" option because he is finished making changes to RACK2. Now the token for RACK2 is held by the Version Control Library.
>
> John decides to fetch SECT1. He chooses the snapshot from above, along with the "Fetch all available Tokens" option. The fetch will retrieve both racks exactly like they were when SECT1 was snapshot, along with both rack tokens. John's copy of RACK2 is now the "current" copy and Tim's changes are lost.

When fetching from the VCL, locks are taken out in the AutoMax library for the appropriate SSRT. For example, if a user is fetching a section, a lock will be taken out in the AutoMax library for that section. No one else can modify the section until the first user's fetch is complete. If the "Fetch Latest" option is being used, locks are also taken out in the VCL so that no one can snapshot anything and modify the "latest" version.

Appendix Q

AutoMax Version Update Utility

AutoMax Executive V3.3 (and later) creates a DPS drive parameter database for each rack when it is added to a system. However, prior to version 3.3, the AutoMax Programming Executive did not create parameter databases. The Version Update utility will add parameter databases to racks created with versions 3.0-3.2 of the AutoMax Programming Executive software.

The Version Update utility works at the AutoMax library level. It will create parameter databases for all the racks in the systems contained in the AutoMax libraries you designate. The update utility will have no effect on AutoMax Executive V3.3 (and later) systems in these libraries. Use the following procedure to create parameter databases.

Step 1. Make backup copies of all AutoMax libraries that you want to update.

Step 2. Access the Windows Program Manager.

Step 3. Access the Resource Group and select **AMXUPDAT.EXE.** A dialog box will be displayed that warns that all libraries that are to be updated should be backed up first. Select OK to display the AutoMax Version Update dialog box.

Step 4. Enter the drive letter and the names of one or more libraries that contain the AutoMax systems you want to update.

Step 5. Select OK to begin updating the systems in the designated libraries. You will receive a message when the procedure is completed.

Step 6. Repeat steps 4 and 5 to update additional libraries or select Cancel to return to the Program Manager.

Appendix R

New Features in This Release

Version 3.x of the AutoMax Programming Executive software contains the following new features.

1. Support for the M/N 57C328 Remote I/O Head. The M/N 57C328 Remote I/O Head is identical to the M/N 57C330 Remote I/O Head, except that you an mix inputs and outputs in the same rail. Each port on the new Remote I/O Head contains separate address spaces for inputs and outputs. Like the M/N 57C330 Remote I/O Head, it can control up to 4 digital rails, analog I/O Rails, or Local I/O Heads. Each Local Head can control up to 4 AutoMate digital rails. Therefore, a Remote I/O Head with 4 Local Heads connected to it can control a maximum of 16 digital rails. See the AutoMax Remote I/O Communications instruction manual for a complete description of the M/N 57C328 Remote I/O Head.

Appendix S

What Can Go Wrong When Loading Tasks and Files

This appendix describes some problems you might encounter when loading tasks and files to the rack.

S.1 Error Messages on the Personal Computer Screen

Error messages that appear on the personal computer screen are usually specific enough to help you pinpoint any problems you are experiencing. In most cases error messages are also specific to the operation that was in process when the error occurred. When you see an error message on the screen during any loading operation, first write down the message. Then, after checking to make sure your connection to the rack is good, and that you are loading the correct file to the designated Processor or UDC module, re-try the operation. Generally, it is best to load every file (except operating system files, which are loaded separately) and task to the rack at one time using the LOAD ALL option.

This section will describe errors that might be difficult to resolve using error messages only. The section is organized by when the error could occur. However, if you load tasks and files to the rack using the LOAD ALL command, you might see any of these errors during the procedure.

S.1.1 Errors that Occur When Loading an AutoMax Processor Operating System or UDC Operating System

In most cases, errors that occur when loading an operating system to the rack are due to improper cabling or high Processor or UDC utilization (a combination of large tasks executing very fast, accompanied by error code “00” on a Processor faceplate). Always make sure that you have a good connection between the personal computer and the rack before trying to communicate with the rack. In addition, it is good practice to stop any tasks that are running in the rack (using the STOP ALL command) before trying to re-load an operating system to the rack. In most cases, re-trying the loading operation will be successful. In rare cases, it may be necessary to replace a Processor module or UDC module in the rack.

S.1.2 Errors that Occur When Loading a Rack Configuration or Drive Parameter Object File

Errors that can occur when loading a rack configuration are usually due to a mismatch between the actual hardware in the rack and what is described in the configuration. For example, it is possible that the configuration describes a hardware module that was mistakenly placed into the wrong slot in the rack, or one that was not installed in the rack at all. It is also possible that the configuration file or parameter object file is actually meant for

another rack. Error messages displayed while loading a UDC task may reference PMI tunables, task tick rates, or a "PMI OS." These error messages indicate a problem in UDC tasks or the UDC operating system.

S.1.3 Errors that Occur When Loading Tasks

Errors that can occur when loading tasks are usually specific enough to pinpoint the problem. If a particular task is specified in the error message, contact your software supplier. Error messages displayed while loading a parameter object file may reference PMI tunables, task tick rates, or a "PMI OS." These error messages indicate a problem in UDC tasks or the UDC parameter object file. Other errors that may occur:

"Task not found"

- attempting to load a task that does not exist in the specified directory on the personal computer

"Task not installed"

- either a task or the rack configuration file has not been loaded into the rack successfully

"Critical task cannot be stopped by itself"

- only a STOP ALL command can stop this task, which you are trying to stop individually

"Task already exists in chassis" and "Given UDC drive already contains a task"

- warns you that if you continue with the loading operation, you will over-write an existing task with the same name

"Invalid data type in symbol table" or "Invalid control type in symbol table"

- there is an error in the configuration file in this rack; it is also possible that the wrong configuration file has been loaded or a wrong task has been loaded

"Too many tunables"

- too many tunable variables defined in a UDC task

"No symbol table present" or "Array name not found"

- error in the configuration file in this rack

S.1.4 Errors that Can Occur in More than One Type of Loading

If an error message indicates insufficient memory on a Processor, the configuration and/or tasks are too large for the amount of memory available on a Processor. Note that this error may occur even if there is technically enough memory (in kilobytes) on the Processor. If tasks have been deleted from the rack or loaded to the rack numerous times individually, the memory available may be fragmented into many small areas, none of which are large enough to store the task or configuration file.

If you see the error **"No common storage allocated,"** there has been an error in the sequence of loading. In this case, you must load (or re-load) the rack configuration to the Processor module. It is very important to load the files in this order if you are loading individual files. Alternately, you can choose the LOAD ALL command

and the Programming Executive software will automatically load the files in the correct sequence.

S.2 Processor or Universal Drive Controller Module LEDs Displaying Errors

If the Processor module OK LED is off or the two seven-segment LEDs on the Processor faceplate display a code, see instruction manual J-3650, which describes the AutoMax Processor module. If the seven-segment LEDs display “LO” (reading top to bottom), this simply means that you need to load the operating system to the Processor.

For racks containing a Universal Drive Controller module, the “OS OK” LED on the module will turn on only after it has received its operating system. See instruction manual S-3007 for more information about the Universal Drive Controller module.

S.3 Loss of Task Modifications Made On-Line

If you or someone else makes changes to tunable variables in existing application tasks in the rack, or makes changes to Ladder Logic/PC tasks on the rack, you must save those tasks back from the rack to the personal computer. This will ensure that you have a copy of the same task on both the personal computer and the rack.

If you do not save tasks back from the rack to the personal computer, and a situation arises that requires you to re-load all tasks to the rack, the version of each task that does exist on the personal computer will write over the version in the rack when all tasks are loaded. If the version of the task on the personal computer is not the most up-to-date version, that is, you or someone else made the changes described above to tasks in the rack, the changes will be lost.

S.4 Tasks in the Rack Do Not Go into Run

Tasks in the rack will go into run, i.e., execute the instructions stored in the task, when they are put into run by the operator using the RUN command. The status (run, stop, or error) of all tasks in the rack can be accessed from the ON LINE menu display. If tasks are not running and there is no error in the rack, you can use the RUN command to start a task or all tasks in the rack.

The nature of the application may require that tasks go back into run after a power cycle (power going off, then on). This is made possible by using the AUTO-RUN software switch. If AUTO-RUN is enabled when power is turned off and then turned on again, all tasks that were in run when power went off will go into run again, assuming there are no errors in the rack.

If AUTO-RUN is enabled and tasks do not automatically go into run when power is turned back on, there is an error in the rack.

RELIANCE CONTROLS
DOCUMENTATION IMPROVEMENT FORM

Document Number: _____

Page Number(s): _____

Comments: *(Please give chapters, page numbers or specific paragraphs that the change will affect. Include markeups from the document or attach additional pages if necessary.)*

What will this improvement suggestion provide? _____

Originator: _____ City: _____ State: _____ ZIP: _____

Company: _____ Phone: (____) _____

Address: _____ Date: _____

Technical Writing Internal Use:	**Follow-Up Action:**
Writer: _____	Date: _____

Thank you for your comments . . .

RE 1857LC Printed in U.S.A.

RELIANCE ELECTRIC

For additional information

1 Allen-Bradley Drive
Mayfield Heights, Ohio 44124 USA
Tel: (800) 241-2886 or (440) 646-3599
http://www.reliance.com/automax

www.rockwellautomation.com

Corporate Headquarters
Rockwell Automation, 777 East Wisconsin Avenue, Suite 1400, Milwaukee, WI, 53202-5302 USA, Tel: (1) 414.212.5200, Fax: (1) 414.212.5201

Headquarters for Allen-Bradley Products, Rockwell Software Products and Global Manufacturing Solutions
Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation SA/NV, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, 27/F Citicorp Centre, 18 Whitfield Road, Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Headquarters for Dodge and Reliance Electric Products
Americas: Rockwell Automation, 6040 Ponders Court, Greenville, SC 29615-4617 USA, Tel: (1) 864.297.4800, Fax: (1) 864.281.2433
Europe/Middle East/Africa: Rockwell Automation, Brühlstraße 22, D-74834 Elztal-Dallau, Germany, Tel: (49) 6261 9410, Fax: (49) 6261 17741
Asia Pacific: Rockwell Automation, 55 Newton Road, #11-01/02 Revenue House, Singapore 307987, Tel: (65) 6356-9077, Fax: (65) 6356-9011

Publication J2-3106-1 - April 1998