

TABLE OF CONTENTS

PREFACE	1-1
MANUAL OBJECTIVES AND READER ASSUMPTIONS	1-1
STRUCTURE OF THE DOCUMENT	1-1
INTRODUCTION TO SYCOM	2-1
GENERAL	2-1
SCOPE AND OBJECTIVES	2-1
FUNCTIONS PERFORMED	2-2
BASIC FEATURES	2-2
LINE THROUGHPUT	2-3
ERROR DETECTION AND RECOVERY	2-4
LIMITATIONS	2-4
NOTICE	2-5
ASSOCIATED DOCUMENTATION	2-5
CONCEPTS AND FACILITIES	3-1
GENERAL	3-1
SYSTEM CONFIGURATION	3-1
SYSTEM COMMUNICATIONS	3-2
POINT-TO-POINT SWITCHED NETWORK	3-3
POINT-TO-POINT NON-SWITCHED NETWORK	3-4
COMMUNICATIONS PROCEDURE	3-5
USER SYSTEM - SERVER SYSTEM RELATIONSHIP	3-6
SYCOM AND ITS ENVIRONMENT	3-8
HARDWARE REQUIREMENTS	3-8
MEMORY REQUIREMENTS	3-8
SOFTWARE REQUIREMENTS	3-9
MASTER CONTROL PROGRAM (MCP)	3-9
SYSTEM/MAKEUSER PROGRAM	3-9
NETWORK CONTROLLER PROGRAM	3-10
SOFTWARE OPTIONS	3-10
SUPERVISORY MESSAGE CONTROL SYSTEM (SMCS)	3-10
APPLICATION PROGRAMS	3-11
FUNCTIONAL DESCRIPTION	4-1
GENERAL	4-1
STANDARD RJE FUNCTIONS	4-1
FILE TRANSFERS	4-3
MAINTAINING INTEGRITY OF FILE CHARACTERISTICS	4-4
VIRTUAL TERMINALS	4-5
TERMINAL-TO-PROGRAM COMMUNICATIONS	4-6
The User System	4-9
The Server System	4-10
REMOTE SUPERVISORY CONSOLE (RSC)	4-11
PROGRAM-TO-SYCOM COMMUNICATION	4-13
PROGRAM-TO-PROGRAM COMMUNICATIONS	4-15
NETWORK CONTROL	4-17
CONVERSATIONAL COMMUNICATION PROCEDURE	4-17
POINT-TO-POINT CONTENTION PROCEDURE	4-18
TRANSMISSION OF PACKED DATA	4-19
NETWORK CONTROLLER	4-19

CONFIGURATIONS AND SOFTWARE ASSOCIATIONS	4-20
LEVEL 1 SYSTEM	4-20
LEVEL 2 SYSTEM	4-22
LEVEL 3 SYSTEM	4-24
MESSAGE PATH ASSIGNMENTS	4-26
FOR THE LEVEL 1 SYSTEM	4-26
FOR THE LEVEL 2 SYSTEM	4-30
FOR THE LEVEL 3 SYSTEM	4-32
QUEUE DEPTH CONTROL	4-35
SYSTEM FILES	4-35
MASTER CONTROL PROGRAM (MCP) FILES	4-36
(SYSTEM)/USERCODE File	4-36
NETWORK CONTROLLER FILES	4-36
Network Information (NIF) File	4-36
SYCOM FILES	4-36
Peripheral Files	4-36
Remote Files	4-37
Queue Files	4-38
RESTRICTIONS	4-38
OPERATOR COMMANDS	5-1
GENERAL	5-1
INPUT COMMANDS	5-1
COMMAND SYNTAX CONVENTIONS	5-2
Notation used in model format	5-2
Punctuation Rules	5-3
Comments	5-3
COMMAND RELATED ERROR MESSAGES	5-4
ABORT	5-5
ACD	5-8
BLFFER	5-13
BYE	5-14
CL (Clear Device)	5-15
CONNECT	5-16
CCPY	5-17
CP NR (Card Punch Not Ready)	5-22
DEBUG	5-23
DISCONNECT	5-24
EOJ	5-25
FETCH	5-26
GROUPSIZE	5-30
LIST	5-31
LCG	5-32
LP	5-33
OFFLINE	5-34
ONLINE	5-35
PHONE	5-36
PROGRAM	5-38
PUT	5-39
QS (Queue Size)	5-43
READ	5-44
RESTART	5-46
RY (Ready Device)	5-48
SIGNAL	5-49
SIZE	5-50

STOP	5-51
SV (Save Device)	5-52
SYSTEMID	5-53
TRACE	5-54
USER	5-55
WAIT	5-56
WHAT	5-57
SMCS Command String from User-System	5-58
SMCS MCP Command String from User-System	5-59
SMCS Command String from Server-System	5-60
SMCS MCP Command String from Server-System	5-61
OPERATING PROCEDURES	6-1
GENERAL	6-1
SYSTEM OPERATING PROCEDURES	6-1
PRELIMINARY PROCEDURES	6-1
Program Switches	6-2
Workfile Naming Conventions	6-4
Creating the SYCOM Parameters File	6-5
Preparing the NEWPARAMS Card File	6-7
Preparing the NEWPARAMS Disk File	6-7
Entering the Parameters Through the ODT	6-9
Creation of the (SYSTEM)/USERCODE File	6-10
Setup MCP Program Name Table	6-10
Assigning Program Priorities	6-10
Execution of SYCOM; Without SMCS	6-11
Execution of SYCOM; With SMCS	6-13
Common System Operations	6-13
SMCS Signal Characters	6-14
Entering the Parameters	6-14
BUFFER Parameter	6-15
DEBUG Parameter	6-15
DEFAULTPSEUDOSIGNAL	6-15
END Parameter	6-15
GROUPSIZE	6-16
INBUFFERS	6-16
NCFS Parameter	6-16
NOFTP Parameter	6-16
NCPROGRAM	6-16
NOTRACE	6-17
NOWAIT	6-17
OUTBUFFERS	6-17
PROGRAM Parameter	6-17
QUEUESIZE Parameter	6-17
SYSTEMID Parameter	6-18
TIMEOUTS Parameter	6-18
TRACE Parameter	6-18
TRACEAREAS	6-18
TRACEBA Parameter	6-18
TRACERB Parameter	6-19
WAIT Parameter	6-19
SYCOM Commands from the ODT	7-1
SYCOM PROGRAM SHUTDOWN	7-2
PRINTING THE TRACE FILE	7-3
REACTIONARY HINTS AND POINTERS	7-3

TERMINAL PROCEDURES	7-5
TERMINAL FUNCTIONS	7-5
TERMINAL OPERATIONS (SYCOM CONTROLLED)	7-5
Example Interactive Terminal Session	7-5
Terminal Operations (SMCS Controlled)	7-8
GENERATING NETWORK CONTROLLERS	8-1
GENERAL	8-1
DETERMINING THE SPECIFICATIONS	8-1
OPERATIONAL PROCEDURES	8-1
PREPARATION OF INPUT	8-1
COMPILATION PROCESS	8-2
PRINT THE USER/NCNIF FILE	8-2
SAMPLE NDL SOURCE FILE (LOCAL USER SYSTEM)	8-3
SAMPLE NDL SOURCE FILE (REMOTE SERVER SYSTEM)	8-6
OUTPUT AND ERROR MESSAGES	A-1
GENERAL	A-1
MASTER CONTROL PROGRAM MESSAGES	A-1
NETWORK CONTROLLER ERROR MESSAGES	A-1
SYCOM ERROR MESSAGES	A-2
CCMMAND INFORMATIONAL/ERROR MESSAGES	A-2
ABORT	A-3
BUFFER	A-3
CL	A-3
CONNECT	A-3
COPY	A-4
DEBUG	A-7
DISCONNECT	A-8
ECJ	A-8
FETCH (also see COPY)	A-8
LIST	A-8
OFFLINE	A-9
PHONE	A-9
PUT (also see COPY)	A-9
QS	A-9
READ	A-10
RESTART	A-10
RY	A-10
SIZE	A-10
STOP	A-11
SV	A-11
TRACE	A-11
USER	A-11
SYSTEM PARAMETER ERROR MESSAGES	A-12
SYSTEM WARNING MESSAGES	A-12
SYSTEM CONDITION MESSAGES	A-14
SYSTEM FATAL MESSAGES	A-17
APPENDIX B: CHARACTER SET	B-1
APPENDIX C: COMMAND SUMMARY	C-1
APPENDIX D: SYNTACTICAL KEYWORD SUMMARY	D-1
APPENDIX E: COMMUNICATIONS CONTROL MESSAGES	E-1
GENERAL	E-1
MESSAGE TYPES	E-1
MESSAGE FORMATS	E-2
BASIC DATA COMMUNICATION MESSAGES	E-2

SYSTEM CONTROL MESSAGE	E-2
ODT MESSAGES	E-3
CARD READER/PUNCH MESSAGE	E-3
LINE PRINTER MESSAGE	E-3
FILE TRANSFER MESSAGE	E-4
VIRTUAL TERMINAL MESSAGE	E-4
FILE SEPARATOR MESSAGES	E-5
PROGRAM-to-PROGRAM MESSAGES	E-6
PROGRAM-to-SYCOM MESSAGES	E-6
APPENDIX F: RESTRICTED MCP COMMAND SUMMARY	F-1
APPENDIX G: PARAMETERS FILE ATTRIBUTES	G-1
RECORD FORMAT	G-1
APPENDIX H: SUMMARY OF FILES	H-1
APPENDIX I: SAMPLE APPLICATION PROGRAMS	I-1
PROGRAM-to-PROGRAM (User System)	I-1
PROGRAM-to-PROGRAM (Server System)	I-2
GLOSSARY OF TERMS AND ACRONYMS	9-1
TERMS	9-1
ACRONYMS	9-13

PREFACE

MANUAL OBJECTIVES AND READER ASSUMPTIONS

The intent of this software operational guide is to enable its user to successfully operate the Systems Communications Module (SYCOM) under control of the B1800/B1700 Master Control Program (MCP). As a means to this objective, this document defines scope and objectives, describes inherent and command driven functions, and presents basic implementation characteristics. Additionally, it provides operating procedures at both the system and terminal level. (Terminal operating procedures are restricted to those capabilities within the scope of SYCOM.)

The Systems Communications Module can be used for a multitude of applications and by users with varying degrees of proficiency in computer operations. Although this guide has a tutorial slant, it does not claim to train operators. The reader is assumed to have a basic knowledge of the B1800/B1700 system and of the Master Control Program (MCP). The reader is also assumed to be familiar with B1800/B1700 operating procedures, to have had experience operating the Operator's Display Terminal (ODT) and peripheral devices, and to have a basic vocabulary of computer related terms. The user need not however, need any knowledge of programming or of the Network Definition Language (NDL).

This guide is designed to be self-contained. However, if the user wishes to have a deeper insight into the system's construction or precise details as to the various internal system functions, a reading of the "Network Definition Language Reference Manual" and the "System Software Operational Guide" is strongly recommended.

STRUCTURE OF THE DOCUMENT

Section 1 presents an overview of the Systems Communications Module along with the functions it performs. This section also identifies any limitations which might be applicable to the usefulness of the system and lists associated documentation.

Section 2 describes concepts and implementation characteristics employed by the Systems Communications Module. Pictorial surveys are presented where applicable to elaborate points of interest.

Section 3 presentation of the Functional Description progresses from basic introductions, to what is involved, through detailed "how it works" explanations.

Section 4 describes directive commands available to the user from either a Work Station (terminal) or the User ODT. With these commands, the user directs SYCOM to perform specific functions to satisfy a user's processing requirements. Examples are offered for each command which includes both command input and reply messages. The commands are presented in alphabetical order for ease of reference. The command language construction is presented in detail; including keyword/parameter semantics.

Section 5 contains operating procedures for a SYCOM system (operating independently or in conjunction with the Supervisory Message Control System (SMCS)) and for local system terminals. Under the heading "System Operating Procedures", useful SYCOM operating procedures, commands, and operational hints/pointers are presented. Under the heading "Terminal Operating Procedures", terminal operational commands and functions are covered which assume either a TD820 or TD830 Video Display Terminal (VDT). However, the system utilizes any Burroughs supported VDT device.

Section 6 presents instructions for the preparation of a Network Controller program required by the SYCOM system. It is basically comprised of a sample NDL coded input file for the Controller's generative process. Various comments are included throughout the sample listing to explain variable parameter data which could be different for each installation. The procedure used to invoke the generative process for an operational Network Controller program tailored to the needs of individual systems is included.

Section 7 contains appendices to this document, each of which is identified below:

Appendix A provides lists of error messages and their meanings. These lists are broken down into two major categories; those error messages which may be displayed in response to commands and those error messages which are displayed on the User ODT which identify a processing exception condition within SYCOM.

Appendix B contains the "character set" defined for terminal input/output and data transmissions.

Appendix C lists "reserved words" recognized by SYCOM, either as a command or command option keyword.

Appendix D contains a "command summary" that lists each command and briefly describes its purpose.

Appendix E provides a list of all control messages used by the communications protocol and includes their associated format attributes.

Appendix F presents a list of restricted MCP commands which are permitted from a terminal or ODT on the User System.

Appendix G describes the PARAMETERS file attributes and details the parameters record format.

Appendix H provides a summary of system files, including names and purposes.

Appendix I contains sample application programs which show the mechanics of interfacing with SYCOM and the system's terminals for program-to-program and terminal-to-program communications. These sample programs can also be utilized in the system test procedures once the appropriate Network Controllers are created and the systems installed.

Section 8 is a glossary of terms and acronyms used throughout this document.

INTRODUCTION TO SYCOM

GENERAL

The Systems Communications Module (SYCOM) program provides the B1800/B1700 user with: 1) the capability to submit jobs to run on another system, 2) to transfer files between two systems, and 3) in general, to interactively obtain services and information from another system with the ability to communicate with interactive programs in a remote system. Job run output can be returned to the submitting system (upon request) or directed to a peripheral device on the processing system itself. Interactive output data is returned to the User System directly, via Network Controllers.

SYCOM along with its associated Network Controller is responsible for all data communications with other systems. This includes performing the necessary functions to establish line connections, receive/transmit data, and execute the correct line termination procedures.

The Systems Communications Module maintains a secondary level of control (MCP has primary control) over local peripheral devices such as the card reader, card punch, line printer, and magnetic tapes. It reads and transfers card, tape, and disk files, accepts input from and displays messages to the Operator's Terminal (ODT) and receives data files/messages for printing, punching, display, or writing to either tape or disk. It provides the mechanism by which a terminal or application program in one system is able to communicate with specific application programs in another system. Program selection is based on the servicing system which dictates the processing requirements (programs) optioned to run in that system.

SCOPE AND OBJECTIVES

The purpose of the Systems Communications Module is to increase the use of data processing facilities (systems) either local or remote to the target system. The scope of SYCOM is to provide all of those features previously afforded by the Remote Job Entry (RJE/TERMINAL) system and additionally provide enhancements with the following expansive communications features: file transfer, terminal-to-program, program-to-program, and program-to-SYCOM. The first two functions assume inter-system communications with the program-to-SYCOM function assuming intra-system communication which is provided by the MCP. The Systems Communications Module with these encompassing capabilities, considerably expands the flexibility of a RJE Station; herceforth referred to as a "User System".

In addition to the above design objectives, several mandatory objectives have been specified. They are: 1) Retain compatibility with the standard Burroughs data communications line discipline and message format conventions and 2) maintain standard Remote Job Entry (RJE) command compatibility with existing RJE program products for Burroughs small, medium, and large systems.

With this software system, the objectives are also to provide the user with the most benefits in terms of flexibility and ease of use. Flexibility in that the user is free to design and install system configurations suited to the requirements of the company for distributed processing. For example, a user can install a B1800/B1700 in the same installation to increase/share resources with an existing system; in a remote installation to place some resources in a remote facility; or in a network of systems to provide distributed processing services. Ease of use in that all capabilities of the system can be selectively utilized by any user thru either command initiated or inherent system functions.

FUNCTIONS PERFORMED

Since SYCOM applications imply transmission of large data files between two or more systems, line throughput maximization, error detection, and error recovery are standard features. Following is a comprehensive list of the functions performed; including those that are under user control. They are presented in three groups, Basic, Line Throughput, and Error Detection and Recovery.

BASIC FEATURES

For SYCOM to accomplish the functions generally described for the system, the following basic features are provided; full details are presented in Section II, "Concepts and Facilities".

- A. Input: Introduction of programs from an input device on the local system for compilation and/or execution on a remote system. Introduction of data from an input device on the local system for processing on the remote system.
- B. Command Input: SYCOM commands as well as SMCS/MCP type commands are permitted from the system ODT, terminals, and specifically tailored application programs running in a system.
- C. Output: Dispersion of data and/or programs produced by the remote system to the local systems peripheral output devices, ODT, terminals, and application programs.

- D. **Monitoring:** Monitoring and controlling programs on the remote system via a local system's Remote Supervisory Console (RSC); virtual terminal, ODT, or application program (requires SMCS on remote system).
- E. **File Control:** SYCOM verifies all file transfers between two systems.
- F. **File Transfers:** The transfer of certain types of files between a local system and a remote system; in both directions.
- G. **Program-to-Program:** Support of communication between a program running in a local system and a program running in a remote system. Also, support a local application program communications link to SYCOM itself.
- H. **Virtual Terminals:** Provide a means by which a terminal, ODT, or application program in a local system can become a virtual terminal entity to a remote system. By design, virtual terminals assigned the status of a Remote ODT if appropriately declared as Server Stations in network controller when connected to a SMCS in the remote system.
- I. **Communications:** Supports a full line of communications; asynchronous, synchronous, and direct connect. Line speed is dependent upon Line Adapter capabilities.
- J. **Security:** The SYCOM controls "log-on" security, log-on involves checking a user's security code and granting access to user files for transfer purposes.

LINE THROUGHPUT

Maximization of line throughput is assured by a point-to-point message transfer procedure which:

- A. Specifically acknowledges receipt of a message with the transmission of either another message block or a "numbered acknowledgement" (#ACK) message. This reduces both line time and turnaround time.
- B. Blocks records for transmission, (i.e., sends multiple data records per message block.
- C. Compress and expand blanks in data records. It is common for card input and printer output to have a large number of imbedded blanks. Eliminating transmission of these blanks significantly improves line throughput.

ERROR DETECTION AND RECOVERY

The following features are provided in error detection/recovery areas to assure reliable service to the system user:

- A. A specific numbered acknowledgement (either in the next message or in a #ACK) guards against the possibility of a lost message by providing an unambiguous acknowledgement for the transmitted message block.
- B. The use of modulo 2 by the transmission numbering facility for the detection of duplicate message blocks.
- C. Dynamic reduction (by command) of the transmission message block size provides the potential for reducing error rates and retransmissions.
- D. The convention of different timeout values for local and remote systems alleviate line collision problems.
- E. The message block transfer procedure eliminates the ambiguous use of certain ASCII control characters (for example, "ENQ" is used only during "establishment"; and nowhere else).
- F. Data transmission error detection is provided thru the use of vertical and horizontal parity in the systems hardware.

LIMITATIONS

The Systems Communications Module (SYCOM) is intended to provide a combined system of RJE, file transfer, program-to-SYCOM message transfer and terminal-to-program/program-to-program communication capabilities. To provide these services and still maintain a certain degree of compatibility with existing Remote Job Entry (RJE) systems, certain limitations become inherent to the system. Therefore, applicable limitations are defined as follows:

- A. SYCOM does not, and will not, support Cassette tapes as an optional peripheral device.
- B. In reference to "file transfers", the implementation currently in effect does not save, or forward, symbolic file headers. Therefore, certain file characteristics would be lost on file transfers between unlike systems.
- C. Program-to-Program limitations involve an application program being assigned a "virtual terminal" position which reduces the number of positions available for terminal attachments.
- D. Interactive application programs in a remote system

serving virtual terminals on a local system cannot make use of "tallies" or "toggles".

- E. The implemented version of SYCOM which this document describes, has only a limited "file transfer recovery" capability. That is, any file queued to "PUT" when the inter-system connection is lost, will initiate a file transfer operation from the "beginning-of-file" at SYCOM BOJ after communications have been re-established.
- F. The most noticeable limitation is the fact that SYCOM only supports a single line connection for RJE and File Transfer data communications at any one time.

NOTICE

The specifications of the software system described in this document are subject to change without notice. The availability or performance of some features may depend on a specific configuration of equipment such as larger memory or communication line speed. Customers should consult their Burroughs Sales Representative for details.

ASSOCIATED DOCUMENTATION

Some of the details of the Systems Communications Module system and its composite software elements are contained in other system manuals. Following is a list of associated documents referenced in this guide. Readers and users of this document may refer to any or all of these listed documents for additional information.

<u>Document Title</u>	<u>Identification Number</u>
B1800/B1700 SUPERVISORY MESSAGE CONTROL SYSTEM	P.S. 2219 0482
MESSAGE CONTROL SYSTEM INTERFACE	P.S. 2212 5447
B1700 NETWORK DEFINITION LANGUAGE	P.S. 2212 5223
B1800/B1700 NCL LIBRARY	P.S. 2212 5215
B1700 DATA COMP AUDIT	P.S. 2212 5421
B1800/B1700 SYSTEMS SYSTEM SOFTWARE OPERATIONAL GUIDE	1068731
B1700 SYSTEM NETWORK DEFINITION LANGUAGE (NDL) REFERENCE MANUAL	1073715
B1700 SYSTEMS DATA COMMUNICATIONS INFORMATION MANUAL	1089992
BURROUGHS CORPORATE DATA COMMUNICATIONS STANDARDS	1284 9006
B1800/B1700 SYSTEMS REMOTE JOB ENTRY (RJE) REFERENCE MANUAL	1090602
B1800/B1700 SUPERVISORY MESSAGE CONTROL SYSTEM SOFTWARE OPERATIONAL GUIDE	Preliminary

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

2-6
COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
P. S. 2219 0458 (A)

B1800/B1700 SYSTEMS REMOTE JOB ENTRY (RJE)

1090602

CONCEPTS AND FACILITIES

GENERAL

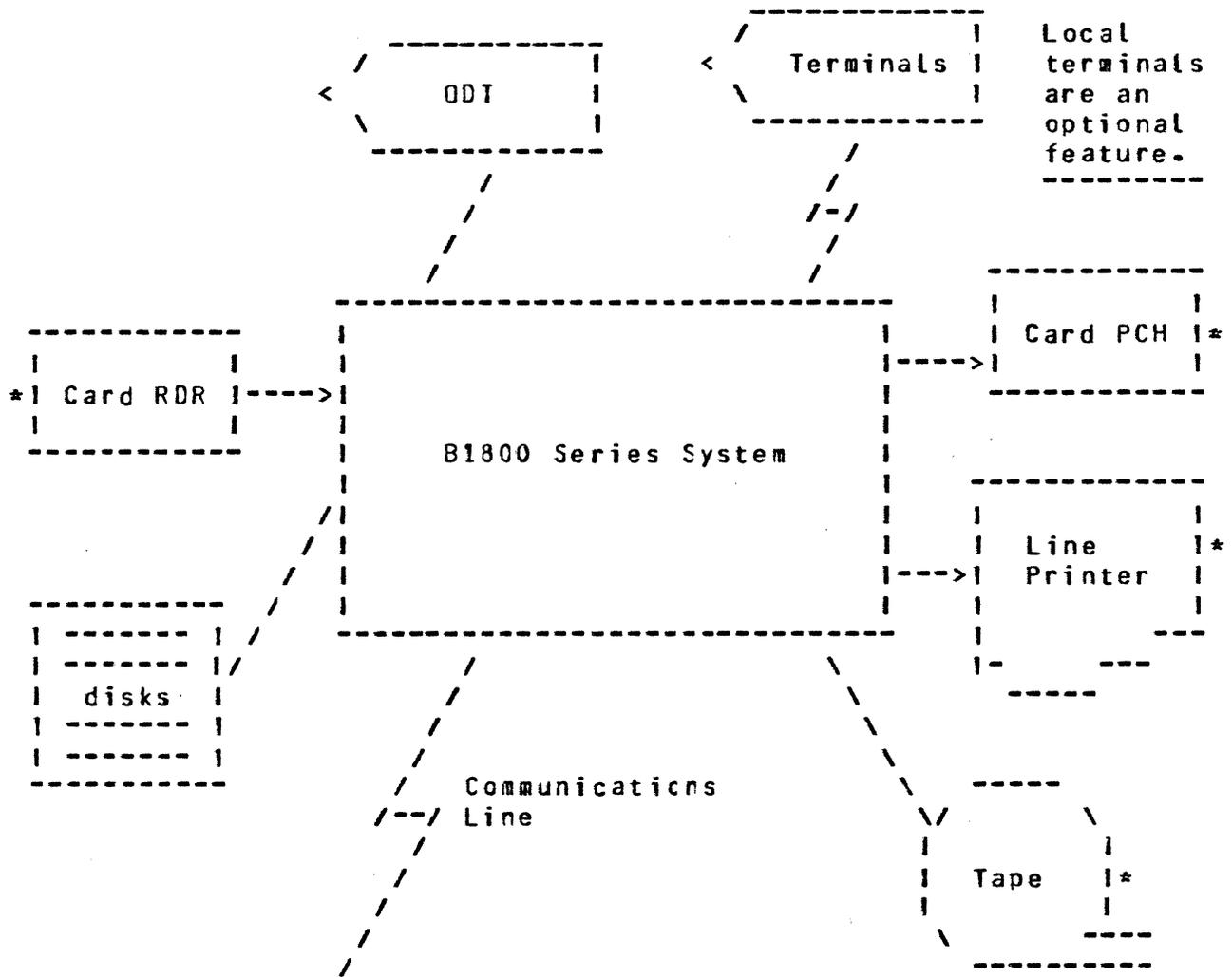
To elaborate upon the design concepts applied to SYCOM, the full range of system facilities are presented in logical progression. First to be covered are actual system hardware configurations. This provides the base for further concept elaborations. Next, descriptions of the data communication networks involved and the relationship of a User System to a Server System are identified. Following this, implementation concepts and functional aspects of the system are described.

SYSTEM CONFIGURATION

A User System oriented to SYCOM utilizes a hardware configuration similar to the one shown in Figure II-1 (next page). A system is usually comprised of the following hardware elements: processor, memory, ODT, disk, plus standard communication controls/adapters. Optionally it can be equipped with an assortment of peripherals as per a user's needs (e.g., card reader, card punch, disk packs, line printer, magnetic tapes, etc. and terminals).

The SYCOM program is responsible for managing availability and use of peripheral devices which are used for SYCOM functions. It also controls all input/output operations for those peripheral devices as well as transmitting and receiving data files. (Data file is a general term used here to encompass all file types acceptable to SYCOM such as data, program source/object, printer backup, etc.)

With this hardware configuration, "all" SYCOM capabilities for a User System are initiated thru the system ODT. An "extended" User System (configuration with local terminals) adds a new dimension to the SYCOM system in that it's capabilities are no longer confined to initiation from the ODT but are now also available to local terminals as well. Commands entered from a system ODT can initiate any SYCOM function, which includes establishing the line connection for virtual terminal communications. Terminals are, however, restricted to an explicit command repertoire. (Details of this concept are presented later in this section.)



(* = optional)

Figure 3.1: User System Configuration

SYSTEM COMMUNICATIONS

Described here are the communications network aspects of a User System and its relationship to a Server System. SYCOM and its associated Network Controller support User Systems using either a point-to-point non-switched or point-to-point switched network to a Server System.

The data communications procedure utilized is a subset of the Burroughs conversational point-to-point line procedure. It is defined in a document entitled "Burroughs Corporate Data Communications Standards"; refer to "associated documentation". That document fully describes the characteristics and functional

attributes for a non-transparent half-duplex mode of data communication to which this system subscribes.

POINT-TO-POINT SWITCHED NETWORK

In a switched network environment, a User System communicates with a Server System without requiring dedicated communication lines; see Figure II-2 (on the following page). The computers are connected by telephone lines to the common carrier exchange serving their respective locations. A temporary connection (data path) is established between the computers only for the period of time in which data transmission is actually required. The connection is established by dialing the telephone number of the Server System. Usually the dialing procedure is manual, however, SYCOM can cause this function to be done automatically upon command directive.

In switched line networks, a line refers to a discrete data path between a Server System and the User System (including terminals) thru the common carrier exchange. The services provided by this common carrier are usually on a time-used basis.

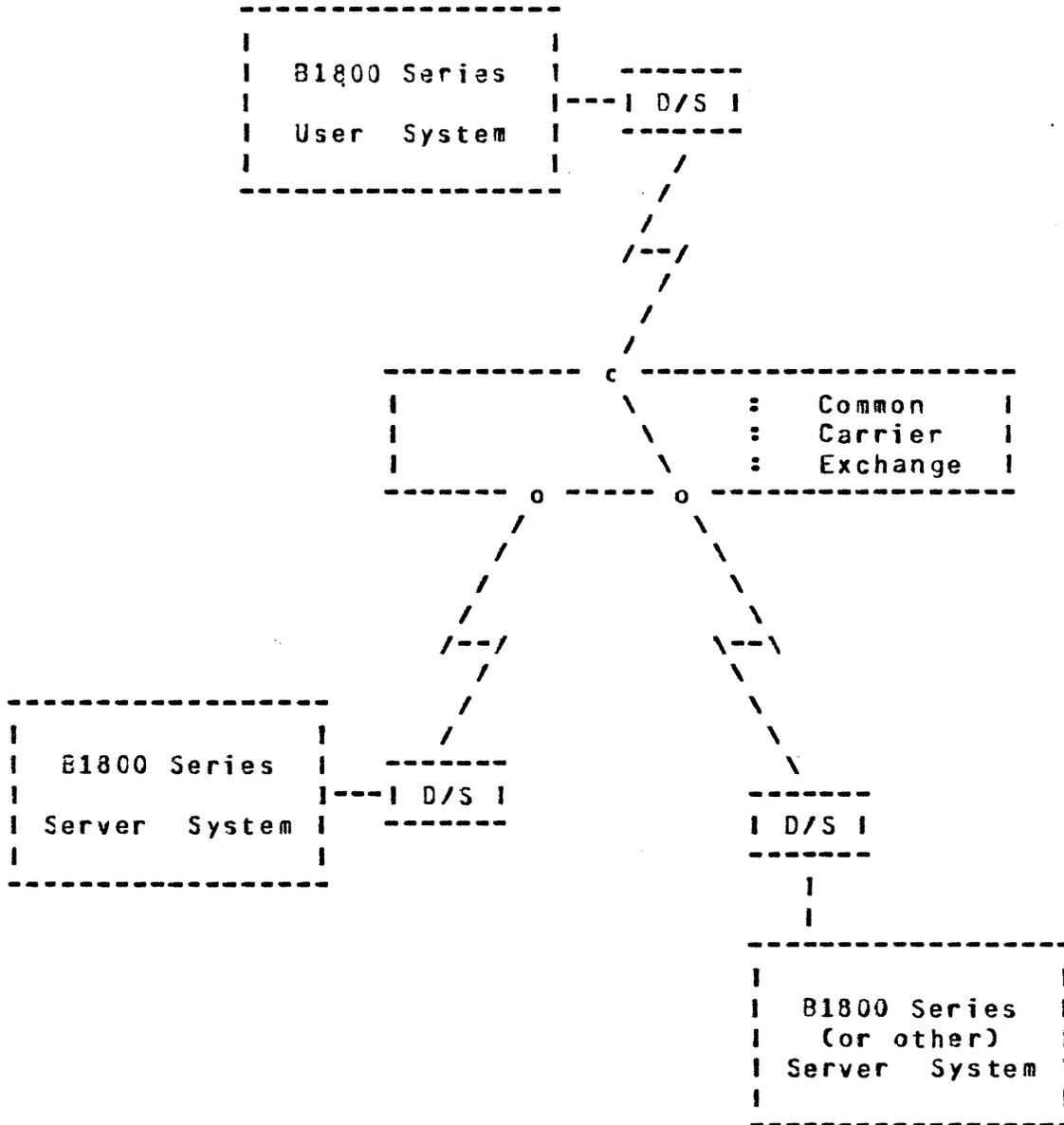


Figure 3-2: Point-to-Point Switched Network

POINT-TO-POINT NON-SWITCHED NETWORK

A point-to-point non-switched network, as shown in Figure II-3 (below), consists of a private or leased communications line that connects the User System with a Server System. In this network, the computers are physically connected, that is, the circuits making up the communications line are permanently established, and data may be transmitted over the line at any time. No dialing is required for this type of network.

Communication lines in a non-switched network are alternately known as private, leased, or dedicated lines. In this document they are referred to as "dedicated lines". These lines may be privately owned (e.g., internal direct connection in the same physical location), or provided by a common carrier on a contract basis. This dedicated connection between two specific locations would provide a line for exclusive system use.

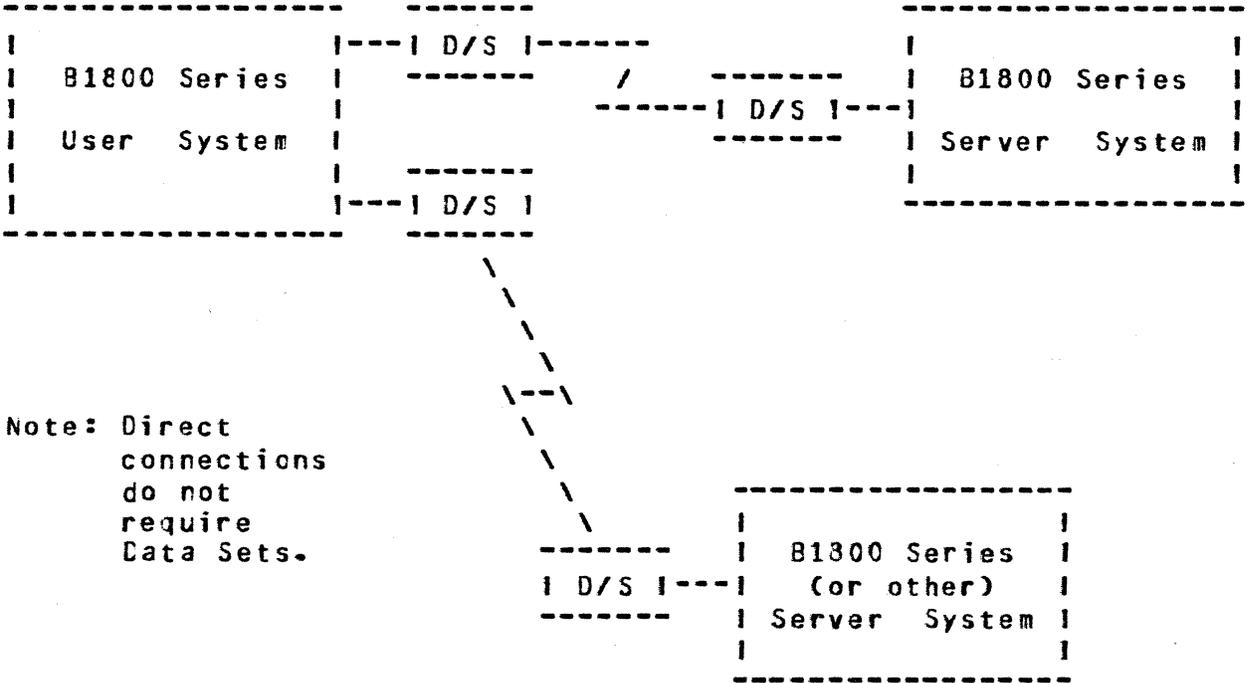


Figure 3-3: Point-to-Point Non-switched Network

COMMUNICATIONS PROCEDURE

With the conversational point-to-point communications procedure assumed by SYCOM, inherent provisions include the capability to "reverse" message block transmissions without having to invoke separate establishment and termination procedures. The essential characteristic of this capability is its ability to specifically "acknowledge" receipt of the previous message block. This is accomplished by including a special acknowledgement character in the header of the "next" message block transmitted. This special character is the Acknowledge Received Message (ARM) character.

The usefulness of the ARM character is twofold: First, it provides a specific acknowledgement to the receipt of a previous message block by utilizing the "numbered ACK" facility. Secondly, when the receiving system receives a message block that it cannot accept (no buffer space or a specific peripheral is not-ready), the ARM character permits an implicit NAK of that

message block. This is achieved by transmitting the NAK in the next outgoing message block.

USER SYSTEM - SERVER SYSTEM RELATIONSHIP

Referring to Figure II-4 (following page), the User System is viewed as the system in the network which serves as a point of data collection and data distribution. For example, RJE type card input, card output, line printer output, and/or terminal input/output messages.

A Server System on the other hand is generally considered to be the data processing system; often referred to as a Host System or Central System. It normally receives large volumes of data from a User System to be processed or program source files to be compiled. In the case of program compilations, the printout and/or program object code files can be transmitted back to the User System (upon request) for printing or possible program execution. Interactive output is returned directly to the initiating program or terminal via the Network Controllers.

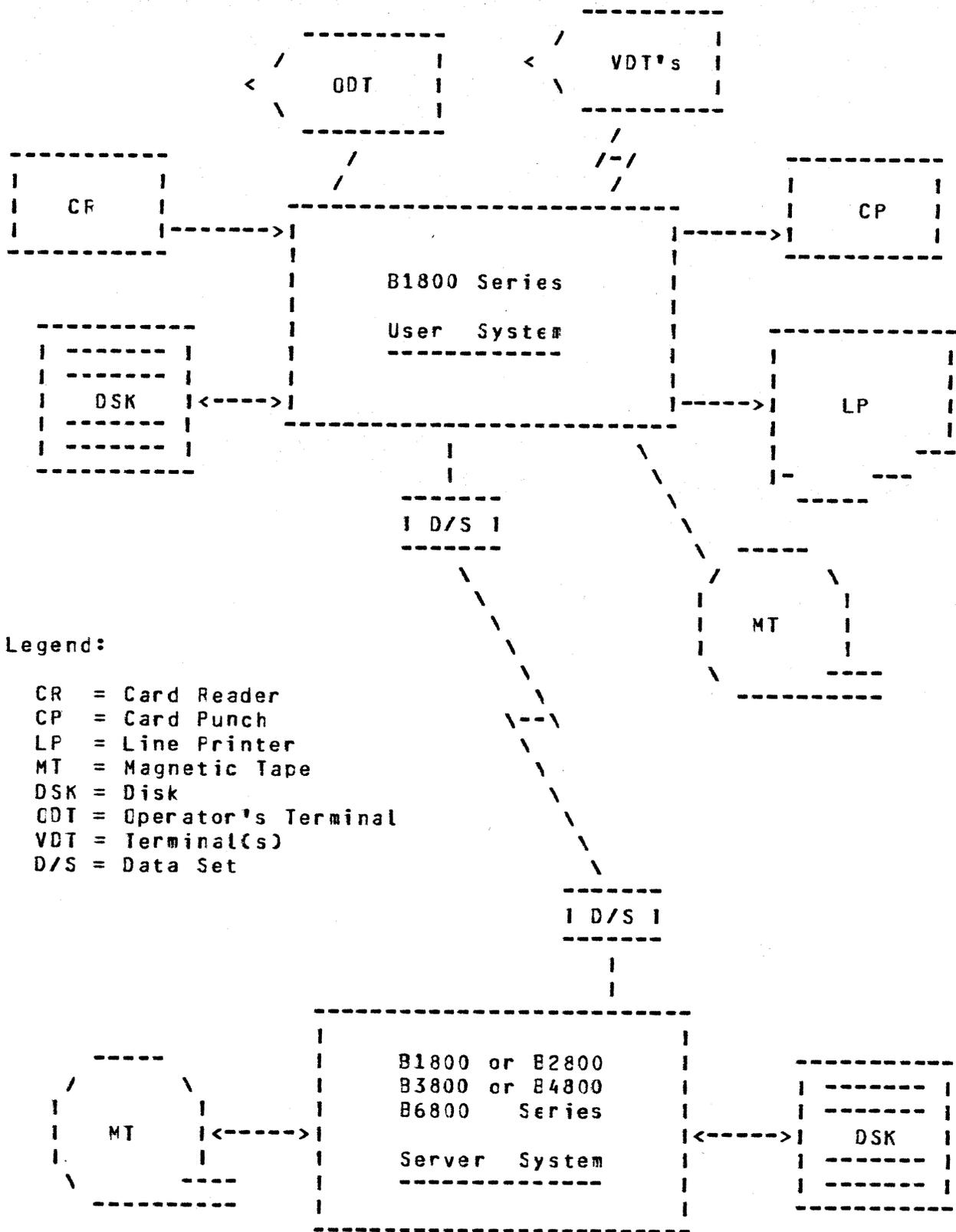


Figure 3-4: User System - Server System Schematic

The actual physical size of a computer however, has relatively little importance when considering a system as a Server System. The B1800 can be either a User or Server System, depending on the scope of the services provided at the time. It can also be, under certain circumstances, both a User System and a Server System at the same time.

SYCOM AND ITS ENVIRONMENT

HARDWARE REQUIREMENTS

The following list of equipment must be present for SYCOM system operations. However, the listed equipment is not dedicated to SYCOM (or supporting system software) and is available for use by other programs.

Hardware Type -----	Usage -----
B1800/B1700 Central Processor	System Control
98 KB Main Memory (minimum)	System Software
Secondary Memory (disk storage) (i.e. head-per-track, pack)	Program/Data Files
120 or 132 Position Printer	Line printer output (optional)
80 or 96 Column Card Reader	Control/RJE Type Input (optional)
80 or 96 Column Card Punch	Control/RJE Type Output (optional)
System ODT (Operator's Display)	MCP/SYCOM/SMCS Operations (optional)
Single or Multi Line Control	System/Terminal Communications
Line Adapters as Required (leased, switched, direct)	System/Terminal Communications
Data Sets and Cables (as required)	System/Terminal Communications
TD820/TD830 Terminals	As Required (optional)

MEMORY REQUIREMENTS

Provided here is a list of System Software elements and their associated memory requirements. They can be used to approximate memory requirements for an interactive B1800/B1700 SYCOM system; operating with or without a Supervisory Message Control System:

- 64.0 KB Master Control Program (MCP).
- 15.0 KB Network Controller (basic).
- 3.9 KB Each additional line in the Controller.
- 16.0 KB Supervisory Message Control System (SMCS).
- 18.0 KB Systems Communications Module (SYCOM).
- 22.0 KB Command AND Edit system (CANDE).
- 2.0 KB Each additional terminal logged-on CANDE.

After determining the basic memory requirements for the system, additional memory must be included for any other programs that are to be executed in conjunction with SYCOM. (See appropriate reference manuals for those memory requirement estimates.)

SOFTWARE REQUIREMENTS

In order for SYCOM to become operational, certain system software elements are required. Although these elements support the operation of SYCOM, they are general purpose and in fact support the entire system. Each software element identified here is individually documented for its capabilities and requirements. Following is a list of those pertinent system software elements:

- Master Control Program (MCP)
- SYSTEM/MAKEUSER program (if "usercodes" are used)
- Network Controller program

MASTER CONTROL PROGRAM (MCP)

The MCP is an operating system which manages the demands and resources of B1800/B1700 systems. It reduces programming efforts by providing a family of commonly needed functions and services. The MCP requires strict user adherence to program structures and procedures in order that there may be communication between the different programs, interpreters, and the MCP itself.

Its primary function is to optimize the productivity of a system. External intervention is held to a minimum, and maximum through-put is achieved by incorporating into the MCP the primary tasks of Input/Output (I/O) control, file handling, multiprocessing, interrupts, memory allocation, application programs, and operator interfaces.

SYSTEM/MAKEUSER PROGRAM

A utility program that is used to create, modify, and access the system's "usercode" file. This file contains all valid system usercode/password combinations. These usercodes-passwords are maintained in a file named "(SYSTEM)/USERCODE". With this file, the MCP can support, and enforce, a file security mechanism for disk files and programs/users that access them.

NETWORK CONTROLLER PROGRAM

The Network Controller is a Network Definition Language (NDL) type program whose prime function is to process and supervise the flow of messages between systems, terminals, and programs. It performs all data communication line discipline functions which are necessary for proper line/terminal control. The Network Controller handles the initiation and completion of communication I/O's and services exception conditions. In order to optimally provide these functions, the Network Controller is configured, by NDL generation, to the specific needs of the system on which it is to be run, (i.e., the number and type lines, number and type terminals, etc.). A product of this generative process is the Network Information File (NIF). This file contains all terminal and communication line attributes as well as logical aspects of the communications network.

Basically, this Controller enables messages to pass between it and SYCOM as well as other terminal oriented programs executing in the system. This is accomplished via MCP supported Remote Files and Queue's along with the Controller's message switching capability. Each program that is to be part of the message flow must contain a "remote file" for input/output message traffic and it must be known to the Network Controller; this includes SYCOM.

SOFTWARE OPTIONS

The following software elements are not required but may be used in conjunction with SYCOM to service a particular user's needs.

- Supervisory Message Control System (SMCS)
- Application Programs
- CANDE Program (multiple-user text editing)
- DMPALL Program

Although the Supervisory Message Control System is not a required system software element for the operation of a SYCOM system, it is definitely suggested for those systems which have terminals. A Terminal in a SYCOM system environment is permitted to initiate "file transfers" or become a "virtual terminal" entity on a Server System. Additionally, these terminals may be utilized for other applications such as data entry and program development. To maintain overall control of the terminals in these varied system environments, the SMCS becomes the controlling element.

SUPERVISORY MESSAGE CONTROL SYSTEM (SMCS)

The Supervisory Message Control System (SMCS) is intended to be the "supervisor" for a data communications software system which includes such Burroughs software elements as "SYCOM", CANDE, and

other online programs of either Burroughs or customer origin.

The SMCS provides to an interactive system, many functions and special features. To fully explore its capabilities, please consult the "SMCS Software Operational Guide". However, the highlights of some of the more pertinent functions are presented here. Relevant to the SYCOM system, SMCS provides some basic message switching (terminal to terminal, terminal to ODT, and ODT to terminal), attachment of terminals to application programs via a remote file, and information about the system's network.

Additionally, it controls terminal log-on security (checking of user security codes and granting access to user files) and approves (or denies) all remote file opens to terminals under its control. Moreover, SMCS provides a certain degree of resource allocation. By this is meant, thru security, job control, remote file control, and error handling, the SMCS can allocate some of the systems resources. For example, for every terminal under its control, SMCS has final control over that device as a resource. It can therefore, temporarily "attach" terminals to SYCOM by means of the SMCS "SIGN ON" facility.

APPLICATION PROGRAMS

Effectively there are three types of application programs useful to the SYCOM system user, they are: 1) Interactive data entry, 2) system-to-system, and 3) command input to SYCOM. Each program type is briefly described here with details of how they actually work presented in the following "Functional Description".

A user can elect to develop an "interactive" application program tailored to specific needs of an installation. These specific needs could include such applicational functions as data entry, information retrieval, or any application desired.

If there is a requirement for an application program in a User System to communicate directly with another application program in a Server System, the capability is there. This communication link is directly thru the Network Controllers and SYCOM has no role in this message traffic. SYCOM does however, initially approve a "virtual terminal" lsn for identification and control purposes. This is a residual capability based on requirements for Network Controllers by SYCOM.

The third type of application program provides the source for a string of SYCOM commands to be performed on an automatic basis. Normally this would be for off-hours operation but could also be utilized for common everyday occurrences of a work flow pattern. (An example of this would be the "SLAVE/BACKUP" program.)

FUNCTIONAL DESCRIPTION

GENERAL

The operation of SYCOM is dependent upon support from a Network Controller program. This Network Controller provides interfacing links between SYCOM and the communications hardware (terminals and other systems). The controller furnishes those services and capabilities which are outlined in general with the preceding subject, "Software Requirements". Throughout this description, Network Controller functions are elaborated upon where it is necessary to maintain continuity of subject matter. Concluding this section, functions of a Network Controller are outlined.

Functional features of the system are described in the following order:

1. Standard RJE
2. File Transfers
3. Virtual Terminals
4. Program-to-Program Communications
5. Network Control

The Virtual Terminal capability contributes Terminal-to-Program, Terminal-to-SYCOM and Remote Supervisory Console communications.

STANDARD RJE FUNCTIONS

The SYCOM program controls, at a secondary level, all typical RJE peripheral devices of a User System. It reads and punches cards, prints output, accepts ODT input, displays ODT output messages, and transfers file/control messages between the User System and a Server System.

Input can be directly from a card reader, from tape or disk thru file equates, or from the ODT. Card reader input is generally associated with RJE batch data transmissions; whereas magnetic tape and disk input more directly relates to file transfers. However, these operations in SYCOM are considered synonymous.

Commands can be alternately input from a specifically designed application program thru the "Program-to-SYCOM" facility. With this mechanism, the RJE input as well as associated output could be automatically controlled on a daily basis without requiring the system operator to enter a long series of commands to SYCOM.

Output can be made directly to a card punch, line printer, disk, magnetic tape, or ODT. However, in the case of a printer or card punch, the data received can be in the form of a "backup file" which is first written to disk. Subsequent MCP commands are entered to cause printing or punching as appropriate.

Furthermore, SYCOM gives certain directives to the Network Controller such as requesting it to establish or re-establish a connection to a Server System. SYCOM also causes the controller to enter an "idle" state while waiting for an incoming connection to be established and to terminate a connection when appropriate.

The transmission of RJE data files to/from a Server System occurs as follows: A user of the User System prepares a card deck for transmission to the Server System. The cards are read, compressed and blocked by SYCOM (upon command) and written into the Network Controller's remote file. (See writeup on "Queue Depth Control".) The controller begins immediately to transmit these card images (message blocks) to the Server System. This process continues until all cards have been read and transmitted. The card file, known as a "job stream", is then under control of the Server System's MCP which processes that job stream as per directives included in that file (compile and/or execute).

All ODT type messages which pertain to that job (BOJ, EOJ, and ODT messages) are transmitted back to the User System for display on the User ODT. Messages can be entered via the ODT to control jobs executing on the Server System or manipulate data files as required; providing a MCS is running in that system. SYCOM accepts these messages and writes them to the Network Controller for transmission to the Server System and subsequent processing.

When the job has finished, any associated line printer and card punch output created at the Server System is transmitted back to the User System. (This is dependent upon the software system in the Server System; HOST/RJE does this automatically where SYCOM does not.) The Network Controller receives these incoming message blocks and forwards them to SYCOM via SYCOM's remote file (input queue). SYCOM then appropriately prints and/or punches this data on a User System peripheral device.

The above described process describes the flow of data between the User System and a Server System; often referred to as a "job stream". Any number of job streams may be initiated at the same time (multiple Server System MCP directives contained in one job file) from a User System. The number of jobs which can or will run on a Server System is limited only to its capabilities to process them.

FILE TRANSFERS

File transfers are supported for both disk and magnetic tape devices. Files may be transferred between any combination of these devices. Magnetic tape transfers are restricted however, in that only one file per tape may be transferred. Tapes created by the LCAD.DUMP, SYSTEM, and COPY functions are not supported. (These tape files are first loaded to disk and then transferred.) The data communications protocol allows transfer of only one file at a time; in each direction.

Whenever SYCOM is operating with a connection to a "ASCII" system such as a B800, the following three File Transfer rules apply:

- 1) Code files are always encoded.
- 2) Data files are not encoded, non-graphic characters are replaced with question marks "?", and
- 3) Non-graphic characters in data files can be transferred intact by specifying "BINARY" in the COPY command string. This forces encoding of the data file specified (all records).

The user initiates a file transfer by entering a "COPY" or "ADD" command to SYCOM (PUT, RPUT, RFETCH and FETCH are acceptable alternates). The syntax allows source and destination systems to be specified (PUT, RPUT, RFETCH and FETCH excluded) for the transfer via attributes. These source and destination system identifiers must be the same as those specified in the "system-id" control message at connection initialization time.

If the specified source system is the User System, the transfer is a "PUT", which is a transfer of a file "to" another system and is based on SYCOM's "SYSTEMID" parameter. When the source system is a Server System, then the transfer is a "FETCH"; which is the transfer of a file "from" another system, also based on SYCOM's "SYSTEMID" parameter.

The actual transfer of a file is initiated by exchange of control information between systems; this process is often referred to as "handshaking". All file transfers occur as "PUT's". That is, if one system wishes to do a "FETCH", it sends a "Fetch message" to the other system. This causes the other system to initiate a "PUT" sequence by sending a "Put Message" to the initiating system. PUT and FETCH messages completely specify both source and destination file attributes. The PUT message elicits a "Put Reply Message", signifying that data transfer may begin. The sender system then sends one or more data message blocks which are eventually followed by an end-of-file message.

Upon completion of the file transfer, the system that received the file informs the originating system of the completion. This notification is made with a "user-destined control message".

If either system wishes to abort an in-process file transfer (for any reason), that system sends either an "Abort Put" or "Abort Fetch" system control message to the other system. An abort is initiated by the appropriate command entered to SYCOM. Alternate "device not ready - suspend" and "device ready - resume" messages are used for flow control at any time for temporary stoppages.

A file transfer request that cannot be processed when entered, because another file transfer is in progress, is queued to disk and processed when next possible. Although there is a maximum depth to this disk queue (see "Restrictions"), it is sufficient to handle all but the most extreme cases. This feature prevents request rejections due to the fact that only one file transfer (in each direction) can be in-process at any given time. Requests are queued on a first come, first serve, basis. This mechanism assures an orderly processing sequence.

MAINTAINING INTEGRITY OF FILE CHARACTERISTICS

A goal of the file transfer mechanism is to maintain as many of a file's characteristics as possible throughout the transfer. For example, if a COBOL source program is transferred, it should be capable of being compiled, without further modification, on the receiving system. Likewise, if a program code file is transferred to a "like system", it should be executable without modification.

Specifically, if a file is transferred between two like systems, even if via one or more unlike systems, all file characteristics must be preserved. To accomplish this, a symbolic version of the file's header (understandable to all systems) is transferred with the file. This symbolic file header is the "PUT-OR-FETCH-RECORD" as defined for SYCOM file transfers. It is prefixed in the data stream by a beginning-of-header control message.

In the receiving system, when a symbolic file header is received, as many as possible of the file characteristics are incorporated into the file being created/received. If the file is transferred to an "unlike" system, the symbolic file header and the original system type are saved with the file, to be forwarded with the file whenever it is again transferred. If the file is transferred to its original system type, the entire symbolic file header is incorporated to create a file identical in characteristics to the original file, and the symbolic header is not saved.

When a system sends one of its own files, it must include in the symbolic file header, enough information for it to recreate that file if returned. If this file is returned, it must be made to

look identical to the original file, with the exception of "file name" (this is conveyed in the symbolic file header).

An unlike system created file may be changed or replaced locally, thus invalidating the saved symbolic file header. In this case, the symbolic file header describes the file according to its new creation system type.

Note: The "symbolic file header" discussed above is not implemented in this version of the system software. It will, however, be available in a later release.

VIRTUAL TERMINALS

To the SYCOM system, a "virtual terminal" can be either a local terminal, the system ODT, or an application program. Therefore, virtual terminals is a logical capability whereby a message path can be established between a terminal, ODT, or program on the User System and an appropriate interactive application program in a Server System. Additionally, utilize the inherent capability of controlling jobs in a Server System with a special MCP command repertoire. Basically, on the Server System end, this becomes a reality thru the capabilities of the SMCS. However, the main inter-system connections are established by SYCOM which makes it possible. Refer to Figure 4-1 for a pictorial overview of a User System with virtual terminals. (Interactive application programs can be any program using a "remote file" but not using "tallies" or "toggles".)

The terminals are virtual in that all User System facilities as well as all Server System facilities are available to the system user. User System commands are local and are therefore, processed within the User System by either SYCOM, SMCS, MCP, or application program. Server System input messages are accepted by the User System and transmitted to the Server System (via the Network Controllers) for processing and subsequent reply messages.

In view of a "virtual terminal" in an interactive environment, Server System commands must be forwarded as soon as possible. To effect this requirement, SYCOM has the capability to "interleave" a "user destined control message" into the data transmission stream immediately if a file transfer is in progress. The control message which contains the virtual terminal message is immediately transmitted on an individual basis to the Server System.

TERMINAL-TO-PROGRAM COMMUNICATIONS

When SYCOM is utilized in a larger system development or data entry environment (with terminals), then the User System depicted in Figure 4-1 on page 4-7 is applicable. In that hardware/software configuration, the complete spectrum of SYCOM capabilities are available to the system user. To be more specific, in addition to the capabilities previously identified, a terminal-to-program capability is available to the terminal user.

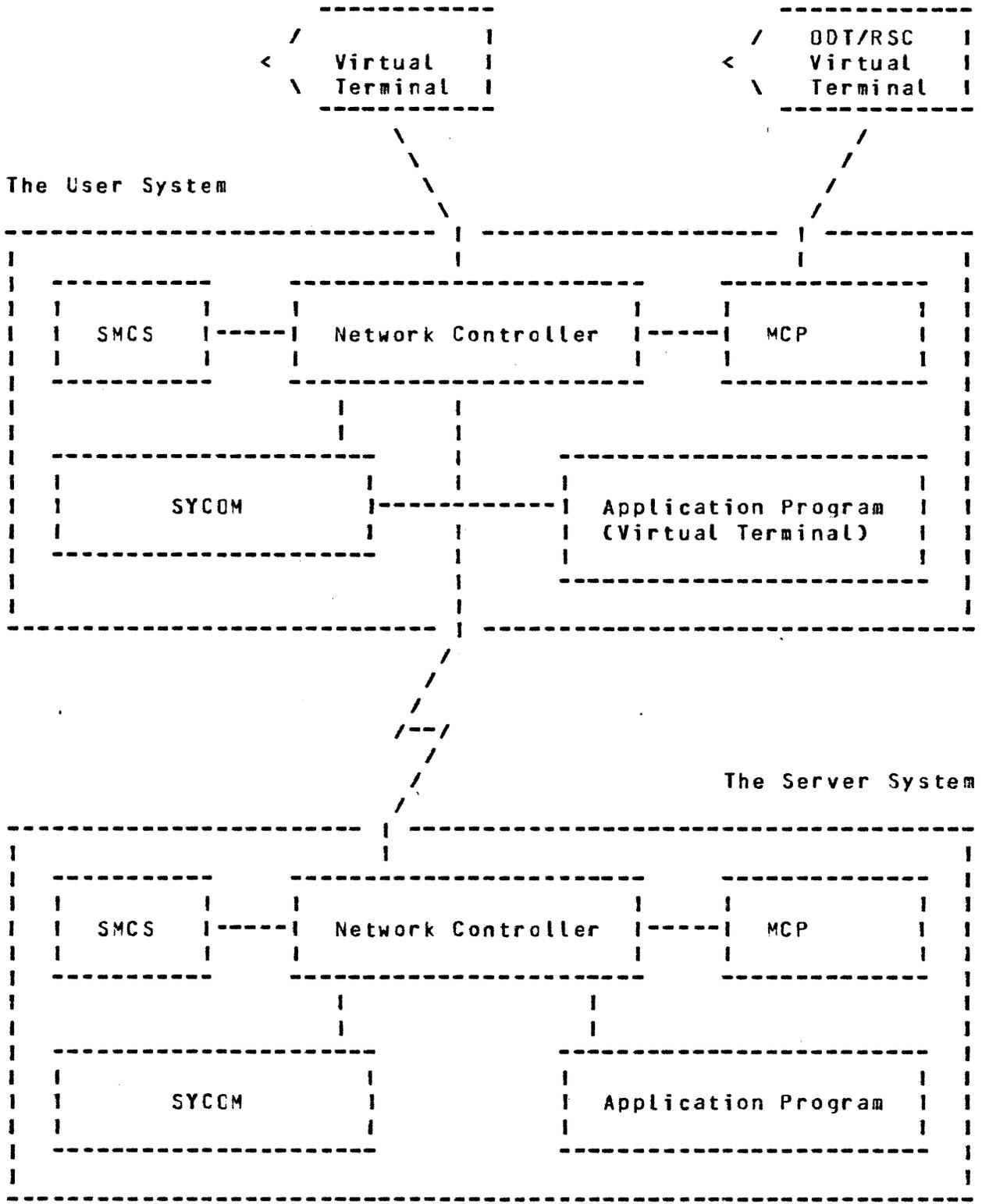


Figure 4-1: Virtual Terminal Schematic

In this environment, SYCOM still communicates directly with the Network Controller for normal communications activity. However, the SMCS maintains elementary control over SYCOM and control over a system's terminals. To clarify this, the SMCS directly causes the execution of SYCOM when a terminal (or RSC) "signs-on" to SYCOM thru the SMCS. A terminal user can then direct commands to SYCOM in the User System or the SMCS in either the User or Server System. Exactly which commands and how the directives are given is covered in Section V, "Terminal Operations". Note that these capabilities are also available to the user who optionally elects not to utilize a SMCS in the system. (Refer to Figures 4-5 and 4-6 for configuration overviews.)

With the introduction of the SMCS software element into a system, all of its commands, functions, and capabilities become available to the terminal user. (Refer to the "SMCS Software Operational Guide" for details and capabilities.)

Before proceeding, it might be helpful to this discussion for the terms User System and Server System to again be clarified. A User System is the computer on which there are terminals/users wishing to engage in interactive computing sessions. The Server System is the computer that supplies those interactive services on a demand basis. (Refer to Figure 3-4 previous section.)

The communications protocol assumed by SYCOM supports User System terminals connected (logically) to an interactive application program on a Server System. Using this protocol minimizes the visibility of interfacing software elements so that to a user, it appears as though the terminal is connected directly to the application program. The terminal is then used as if it was locally attached to that program (see "Message Path Assignments" further on in this section).

Presently available interactive application programs may be used or the user can optionally elect to develop a special application program for terminal and/or other program interaction.

The method for conveying terminal data between two systems makes use of the "device address" field in the control message. This message specifies the terminal with which the message text is associated. The device address in the context is termed "virtual address", and is the name used by both User and Server Systems to reference a particular terminal. This addressing scheme follows:

- A. The device address is comprised of two characters; one upper case and one lower case alphabetic letter.
- B. The User System "transmits" a virtual terminal address as lower case/upper case letters. When it "receives" a response message from the Server System, the terminal address letters are reversed; upper case/lower case.

- C. The Server System "receives" a virtual terminal message with a device address of lower/upper case letters (as transmitted by the User System). The virtual terminal addressing scheme then requires the Server System to "reverse" these device address characters prior to its transmission back to the User System. (These transmit/receive device addresses are specified in the Network Controller's NDL generation process; on "both" systems.)
- D. Assume that a User System supports a connection for five terminals to a Server System. The User System refers to these terminals by virtual terminal (transmit) address:

aA, bA, cA, dA, and eA

The Server System also refers to these same terminals by virtual terminal (transmit) address; but, as the Server Systems role is reversed, its transmit address is also reversed. For example:

Aa, Ab, Ac, Ad, and Ae

The following events represent the terminal-to-program procedure implemented by SYCOM. These events portray an interactive session between a virtual terminal on a User System and an application program on a Server System.

The User System

At initialization time, SYCOM issues a "remote file information" request to obtain, from the Network Controller, the Logical Station Numbers (lsn's) for the virtual terminals assigned to the system. These numbers are placed into a table which maps the actual "lsn" of each virtual terminal into a virtual "lsn" with which SYCOM can reference the corresponding virtual terminal.

When a user wishes a connection to a Server System, a ".CONNECT" command is entered to the User System's SYCOM. SYCOM finds the "next" available entry in its "virtual lsn table" and transmits a "connect message" to the specified Server System. This control message contains a virtual terminal address. Upon receipt of the "connect reply message", SYCOM replies with the results.

During the life of a connection, all data messages entered from a virtual terminal are sent to the Server System for processing. (Excluded are any local SMCS commands which are prefixed with the assigned signal character.) This is accomplished by doing a remote file write to the Controller using the terminal's virtual terminal "lsn". In processing this message, the Controller maps the virtual terminal lsn into the assigned virtual terminal address (two characters). The message is then transmitted to the

Server System, according to SYCOM protocol, with the virtual terminal address. The terminal user can then "sign on" to any application program that is supported in the Server System; providing the capability is inherent in the form of a MCS.

Any incoming message to the Network Controller from a SYCOM line which contains a virtual terminal address (e.g., "Aa", "Ab", etc) is sent to SYCOM. These messages are written into SYCOM's remote file (LINK) by the Controller with a device address appropriate for the terminal. SYCOM upon reading the message, searches the virtual terminal address table for the actual system "lsn" for that terminal. SYCOM then puts the actual "lsn" into the reply message and writes it into the Network Controller's remote file (station queue) for distribution to the originating terminal.

In the case where the virtual terminal is actually an application program, the reply message is sent to that program via the queue mechanism for "program-to-SYCOM" communications.

When the user wishes to disconnect from the Server System, a ".DISCONNECT" command (.BYE command or *OFF command may also be used) is entered to local SYCOM which transmits a "disconnect message" to the Server System, specifying the virtual terminal address to be disconnected. The terminal is notified of this action and is then free to establish another connection or work pattern as desired.

The Server System

The Server System's Network Controller intercepts "connect" and "disconnect" messages (unlike all other control message which are routed to SYCOM). On receipt of a "connect message", the Network Controller verifies that the addressed virtual terminal exists, checks to see if it can support another connection, and that some interactive application program is ready to accept input from it. If all is correct, the Network Controller returns a "connect reply message" indicating successful completion, else the Controller sends back an error response in a "connect reply message".

When the connection is successful, the Server System makes this virtual terminal appear as a terminal local to that system. If the user is "logged-on" to SYCOM, then that user is automatically "logged-on" to the Server System's SMCS.

During the life of the connection, any input data received by the Server System's Network Controller for an enabled virtual station is conveyed to the waiting application program thru its remote file (could be the SMCS). This message transfer transpires in exactly the same manner as for local terminals. Any output data generated by that application program is transmitted back to the User System, with the virtual terminal address specified.

The Server System upon receipt of a "disconnect message", takes necessary action to detach that virtual terminal and respond to the User System with an appropriate "disconnect reply message". If the user was logged-on to the Server System's SMCS, then that user will be automatically "logged-off" the virtual terminal.

REMOTE SUPERVISORY CONSOLE (RSC)

It is an inherent capability of the system to utilize the ODT, an application program, or a terminal in a User System as a "Remote ODT" to a Server System. This only occurs when, as Figure 4-2 shows, a copy of the SMCS is running in the Server System. For this system, the ODT is also referred to as a "Remote Supervisory Console", or simply the RSC. This term was selected to imply capabilities which transcend normal capabilities of a ODT. This interface infers communication (via the SMCS) with a Server System's MCP, hence the term, Remote Supervisory Console.

Messages input from the RSC to SYCOM are grouped into two basic types:-Local SYCOM and Remote MCP. Local SYCOM input includes commands to initiate file transfers, request SYCOM status data, and other such functions. Additionally, respond to conditions within SYCOM's domain that cannot be preprogrammed; such as card reader not ready, clear peripherals, etc..

Remote MCP commands permitted from a RSC are basically directives used for control over the execution of jobs in the Server System; all via the Server System's SMCS. Only certain MCP commands are permitted from a RSC (reference Appendix F, "Restricted MCP Command Summary") when the RSC's "virtual terminal" assignment is not declared as a Remote ODT in the Server System's Network Controller; otherwise, only Remote ODT restrictions apply.

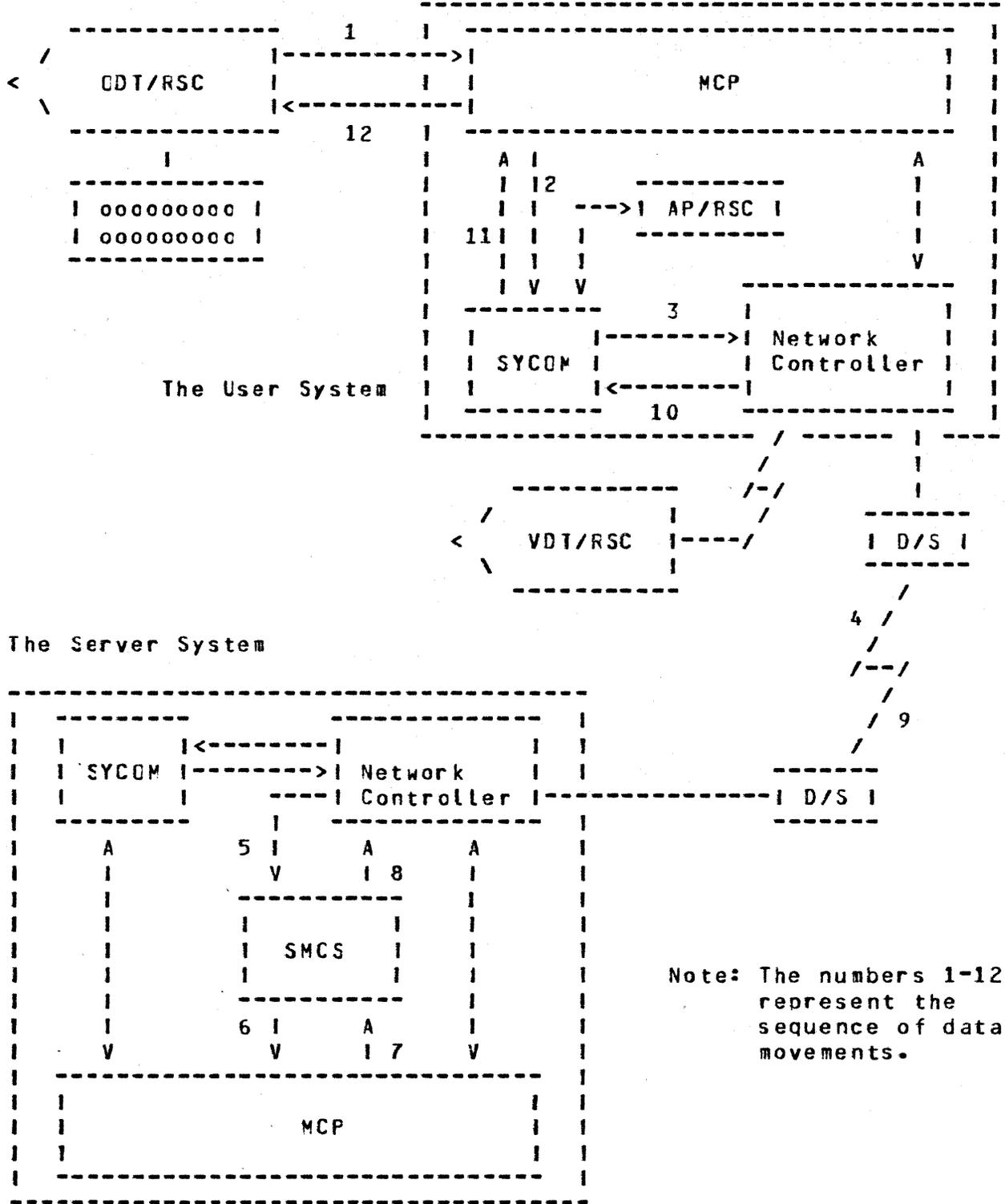


Figure 4-2: RSC Input/Output Message Flow

Output messages for RSC display can be grouped into four types; they are: Local SYCOM, Local MCP, Remote SMCS, and Remote MCP. Local SYCOM output messages are normally responses to commands but can also be error type messages reporting SYCOM problems. Local MCP output would include such messages as "NO PRINTER AVAILABLE" and other such information messages which pertain to operations of a SYCOM system. Remote SMCS messages would be replies to SMCS commands entered for processing by the Server System's SMCS. Remote MCP output messages include ODT type output resulting from jobs run on a Server System. These messages are routed back to the User System's SYCOM program for display.

In the case where the RSC is actually an application program, the response message is directed to that program via SYCOM's "program to SYCOM" queue mechanism.

Referring again to Figure 4-2, assume for example, that a user wanted to execute a job on the Server System. Assume also that the subject program and all data requirements reside in the Server System. The appropriate command string is entered to the User System's SYCOM, via the ODT/RSC. SYCOM accepts this message from the MCP, formats and transmits it to the Server System, via the Network Controller. (The command string could also be entered from either a local terminal or application program; Although these message paths are slightly different, the end results are the same.)

In the Server System, the SMCS receives the MCP command message (from the Network Controller) and "zips" it along to the MCP. The Server System's MCP processes the request as asked. All output messages pertaining to that job are given to the SMCS which in turn initiates transmission to SYCOM in the User System. User SYCOM then displays these messages on its ODT/RSC.

From this we can see SYCOM's involvement and the processing flow involved in this type of operation. All preceding functions described for the RSC are over and above the standard User ODT input/output messages used for normal User System operations.

Since SYCOM is symmetrical, both systems can perform an identical set of functions; providing of course, both systems are of the same series. Actual input and output messages are exemplified in Section V, "System Operating Procedures".

PROGRAM-TO-SYCOM COMMUNICATION

The same range of functions are possible for the Program-to-SYCOM facility as for Terminal-to-Program Communications. This feature is made possible by an adaptation of the Terminal-to-Program scheme.

A mechanism (of implementation dependent nature) is provided to allow an application program in the User System to masquerade as a "terminal". This application program must format messages preceded by the program header to appear as if they were terminal keyboard input. These messages are then written into SYCOM's "SYCOM/OUTQ" queue for processing.

From this point on, at least until a reply message is returned, processing for SYCOM is the same as terminal-to-program commands. When SYCOM produces a reply message, it writes the message into its "SYCOM/INQ" queue file from where the application program in turn reads this queue to obtain the message. These reply messages are formatted to terminal display specifications. Therefore, the application program can either print, store, ignore, parse, or display select message text as appropriate for the program.

The schematic on the following page (Figure 4-3) provides an overview of the mechanics for "Program-to-SYCOM Communications"; Communication between the application program and SYCOM is thru the previously identified pair of "queue files". The application program gives a command to SYCOM by writing into SYCOM's input queue file named "SYCOM/OUTQ". The application program obtains a response message by reading from SYCOM's "SYCOM/INQ" queue file.

Message Formats used for "Program-to-SYCOM" communications are defined in Section VII; "Appendix E". No command restrictions are enforced for this method of SYCOM/system control. However, this feature is controlled by the SYCOM parameter "PROGRAM" and for this function to be available, that parameter "must be" specified when the "SYCOM/PARAMETERS" file is created.

The SYCOM command "PROGRAM" may be used to interrogate the current value of the PROGRAM parameter. If it is entered from the system ODT or a station known to the SMCS as a remote ODT, then the default value of the PROGRAM parameter may be changed during later execution of SYCOM.

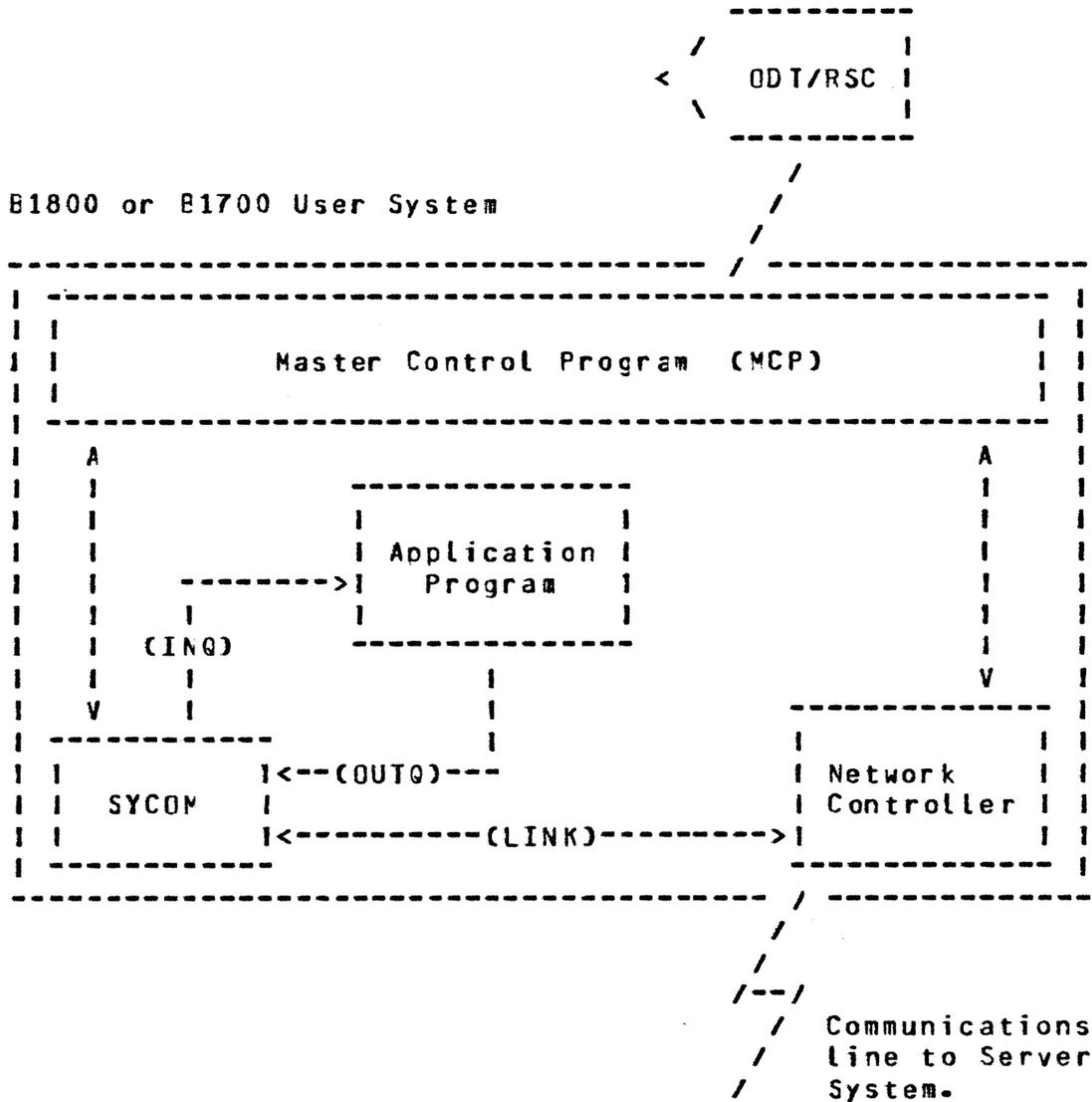


Figure 4-3: Program-to-SYCOM schematic

PROGRAM-TO-PROGRAM COMMUNICATIONS

The program-to-program capability employs an entirely different concept for inter-system communication. As previously discussed, this feature is an inherent by-product of the system. SYCOM itself is in no way involved with the program-to-program message traffic and therefore, does not provide any services to either program. The only exception to this is that SYCOM does "approve" a virtual terminal position for the program communications path.

NETWORK CONTROL

The SYCOM User System is normally considered to be the remote RJE Station and must initiate the transmission of data to the Server System. In a point-to-point non-switched network, this is accomplished automatically by enabling the line adapter and then exchanging handshaking control messages with the Server System.

In a point-to-point switched network, the User System must call the Server System. This is accomplished by either dial-out (automatic dialing by the computer) or by manual dial-up (dialing performed by the system operator). The dialing is done "after" the adapter is enabled. When the line connection is established, the exchanging of handshaking control messages with the Server System is carried out. In a point-to-point non-switched network utilizing a direct connection, no dialing is done, or possible.

Once this communication link has been established with a Server System, line communication is identical for the switched and nonswitched networks. (The User System and the Server System must be compatible as to transmission rate and other such optional hardware capabilities.)

CONVERSATIONAL COMMUNICATION PROCEDURE

The following notes refer to the basic conversational line discipline employed by SYCOM. They apply to a point-to-point circuit, either dedicated or switched:

Message Transfer:

All message blocks are delimited by special characters. Messages begin with a "SOH" character and end with a "BCC" character. The message control format is as follows: SOH, ARM, TM#, STX, DA1, DA2, message text (blocked records), ETX, and finally the BCC.

Retransmission of Message:

If the transmitting system does not receive a reply to its last transmission within a given period of time, the unacknowledged message block is re-transmitted.

Parity or Block-Check Error Handling:

If the receiving system receives a recognizable message, but it contains either a parity or block-check error, it replies with a "NAK". The transmitting system should retransmit the message. This is repeated "n" number of times ("n" being user specified), and if the error persists, either system can initiate error recovery.

Message Idle State:

If the receiving system receives a "good" message, but has no message traffic to send, it responds with an "ACK". If the sending system has more message traffic, it sends another message block. If the sending system has no more message traffic, it proceeds to the "message idle state".

Normal Termination:

The normal termination of a conversation occurs when one system sends an "ACK" (indicating that it has no more message traffic), and the system receiving the "ACK" also has no more message traffic. The system receiving the "ACK" proceeds to the "message idle state".

POINT-TO-POINT CONTENTION PROCEDURE

The essential characteristic of this procedure is that after the end-of-message transfer, as indicated by the transmission of the "DLE-EOT" sequence, both systems are in an idle state. Either system may then bid for master status by sending the "DLE-ENQ" sequence. This is normally initiated by the entry of a command to SYCOM.

Master Status:

If either system is in the idle state and wishes to begin a message transfer, it may bid for master status by sending the "DLE-ENQ" sequence. If the same system had just terminated a message transfer by sending a "DLE-EOT" sequence, then there is a delay created before this system again sends the "DLE-ENQ" sequence.

Begin Message Transfer State:

If the system bidding for master status receives an "ACK" (indicating ready to receive), it may commence message transfer immediately. If the response is "NAK" (not ready to receive), the "DLE-ENQ" sequence may be repeated up to a retry limit of "n" ("n" being user specified), and if no valid response is received, commence error recovery. If the system bidding for master status receives the "DLE-ENQ" sequence (indicating that the other system is bidding at the same time), it abandons its own bid and replies in accordance with its readiness to receive. In order to avoid any line collisions when both systems are bidding at the same time, the no-response time-out values must be set to different values. This is true whether or not the systems are "like" or "unlike".

Termination:

The "DLE-EOT" sequence is transmitted by the system directed to terminate to indicate the end of its session, and if using a switched line, to disconnect the modem.

TRANSMISSION OF PACKED DATA

As in previous versions of this protocol, data records within a message block are separated by "RS" characters. Repeated occurrences of blank characters are compressed out using the "ESC-integer" mechanism. The method of representing non-printable characters has changed with SYCOM's implementation. Printable characters are those listed in "Appendix B, Character Set", with the exception of the "Line Feed" and "BELL" characters.

If the data records contain only printable characters, then the first character of the record "is not" a "DLE" character. If the data record contains any character not in the "character set", then the first character of the record must be a "DLE" character, and the remainder of the record contains translated data. The original data is translated such that every 4-bits is mapped into an 8-bit printable character. (Actual data transmissions occur in ASCII character mode.)

NETWORK CONTROLLER

The Network Controller provides interfaces with the User System's terminals and Server Systems at the hardware level. It is responsible all for direct communication with those terminals and Server Systems. It performs the actual data transmission (both transmitting and receiving) functions between two systems. The Network Controller accepts data message blocks and system control messages from SYCOM and certain application programs (program-to-program) and transmits them to the Server System. In reverse, it receives all messages from a Server System and forwards them as required for handling control and distribution. The SYCOM program relies upon the MCP for the basic function of "message queuing" between it and the Network Controller and certain application programs. This facility provides the message paths between SYCOM, the SMCs, application programs, and the Network Controller.

Following is a normal sequence of events which transpires between SYCOM, the Network Controller, and the system's terminals:

In response to a "poll" by the Network Controller, a terminal transmits a message back to the Network Controller. The Network Controller accepts the message and appends a header, supplying pertinent information about the message, (e.g., terminal type, lsn, etc.). The Network Controller then determines which program to send this message to and writes it into the appropriate program's remote file (input queue). (Assume in this case it is SYCOM.) SYCOM reads its remote file (input queue) and obtains an input message. Should the message be a SYCOM command, it is processed and a reply message created. The reply message is then started down the line by SYCOM when it writes the output message into its remote file to the Network Controller. The Controller

in turn reads the reply message and transmits it (devoid of header of course) to the originating terminal.

The description on the relationships of the SMCS and the Network Controller with SYCOM, which is also considered an application program, apply to any application program. Thus, any application program can enjoy the benefits of a sophisticated terminal network with message/terminal/program management thru the SMCS and Network Controller. In summation, the SMCS is the message controller for the system. It also provides a myriad of functions which direct Network Controller functions as well as special purpose MCP functions.

CONFIGURATIONS AND SOFTWARE ASSOCIATIONS

The Systems Communications Module has been designed to operate in either of three modes:

- Level 1) as an individual entity "without" local terminals
- Level 2) as an individual entity "with" local terminals
- Level 3) in conjunction with the SMCS; also with terminals

Referring to Figures 4-5, 4-6 and 4-7 (which follow), the system configurations with supporting software elements are shown and described.

LEVEL 1 SYSTEM

This system configuration provides all of the SYCOM capabilities previously defined which includes the ODT as a virtual terminal. This is accomplished with the smallest memory size commitment. Although the command repertoire is generally confined to entry from the system ODT, special application programs running in the system can alternately inject commands to SYCOM for processing.

In the User System schematic which follows (Figure 4-5), the communication links between system software elements are shown. Each of these links, "queues" and "remote files", have specific purposes which are briefly identified as follows:

- A. "INC" is a queue file used by SYCOM as output to an application program utilizing the "Program-to-SYCOM" facilities.
- B. "OUTQ" is also a queue file used by SYCOM as input from an application program utilizing the "Program-to-SYCOM" facilities.
- C. "LINK" is a remote file which is used in communications with a remote Server "hcst" System. All RJE and virtual terminal input/output message traffic moves thru this file. This remote file is directly related

to the SYCOM communications line to a remote host.

- C. "RFUSER" is a user defined name for a remote file used for program-to-program I/O message traffic, primarily directly thru the Network Controllers. All messages for this remote file move thru the same communications line as those identified for the "LINK" file.

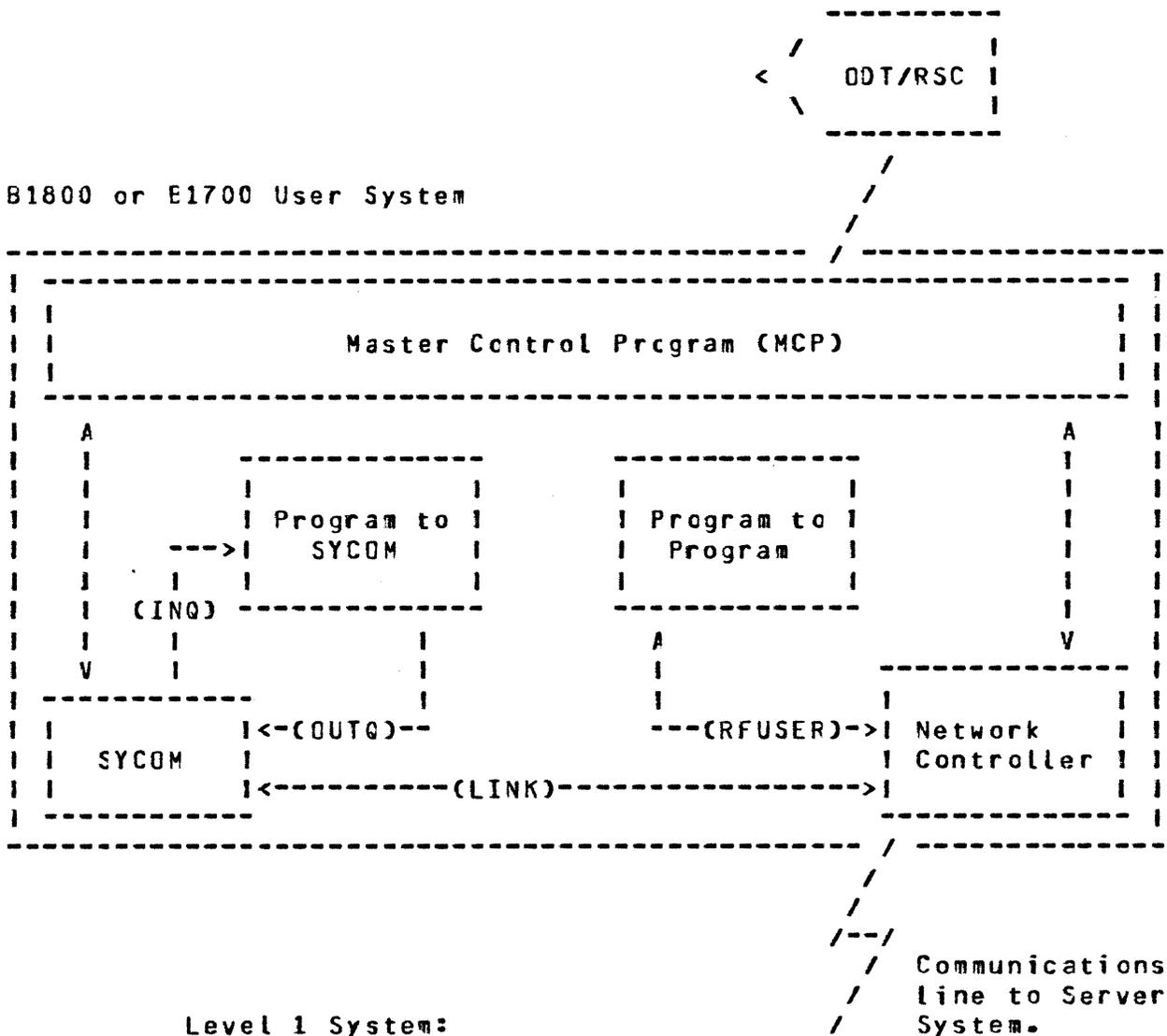


Figure 4-5: User System Schematic

LEVEL 2 SYSTEM

This system configuration also provides all SYCOM services as outlined for the Level 1 System and extends this to include all local terminals. With terminals, SYCOM functions can be invoked from outside the computer room and away from the system ODT. Terminals on the system increase memory usage in proportion to the number of terminals attached to the system and the line or lines for terminals. The memory increase however, will be in the Network Controller program and not SYCOM itself.

Local terminals are exclusively owned and controlled by SYCOM. Although there is no operational SMCS required in a User System, it is "required" in a B1800 series Server System when Remote ODT type activities are needed. The introduction of terminals in this system requires another remote file declared in both SYCOM and the local Network Controller:

- A. "TERMINALS" is the remote file used by SYCOM for local terminal input/output message traffic between it and the Network Controller. As the controller handles all necessary terminal interfacing, SYCOM only reads input from and writes output to this remote file.

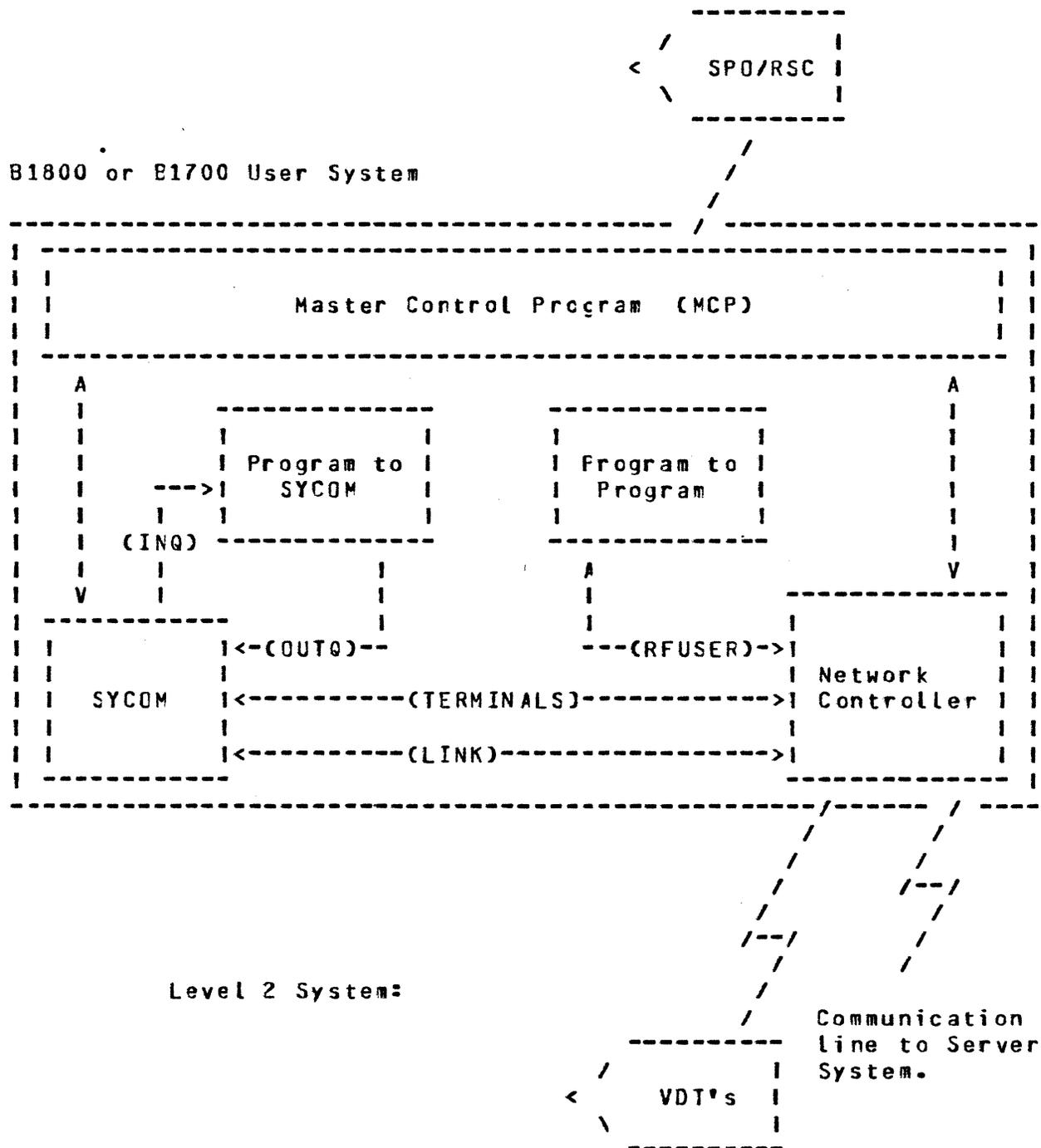


Figure 4-6: User System Schematic (terminals)

LEVEL 3 SYSTEM

Using this configuration, not only are all of the capabilities provided as outlined for both the Level 1 and Level 2 systems, it now provides the SYCOM terminal user with an access to all of the capabilities afforded by the Supervisory Message Control System (SMCS). Memory increases in this configuration are directly related to the SMCS and the Network Controller.

Local terminals in this system configuration are under direct control of the SMCS. Upon command (SMCS "sign on"), a terminal is temporarily "attached" to SYCOM for functional communications. When this SYCOM attachment is no longer needed, the terminal is "detached" from SYCOM by entering "<smcs signal character> OFF or .BYE"; This command is passed on to the SMCS by the Network Controller which initiates the actual detachment and again assumes control of the terminal. As such, an additional terminal I/O link is needed. Also, in addition to this link, SYCOM's remote file "TERMINALS" is effected somewhat. Both are described as follows:

- A. "MCSREMOTE" is a remote file used by the SMCS as a link to the system's terminals. All terminal I/O message traffic between SMCS and the Network Controller passes thru this file. During the time a terminal is "signed-on" to SYCOM, certain messages are "switched" to the SMCS for processing (those messages prefixed with a SMCS "signal character").
- B. "TERMINALS" is SYCOM's remote file for terminal input/output message traffic between it and the Controller. The slight difference being that terminal input/output only passes thru this file for those terminals that are "signed-on" to SYCOM.

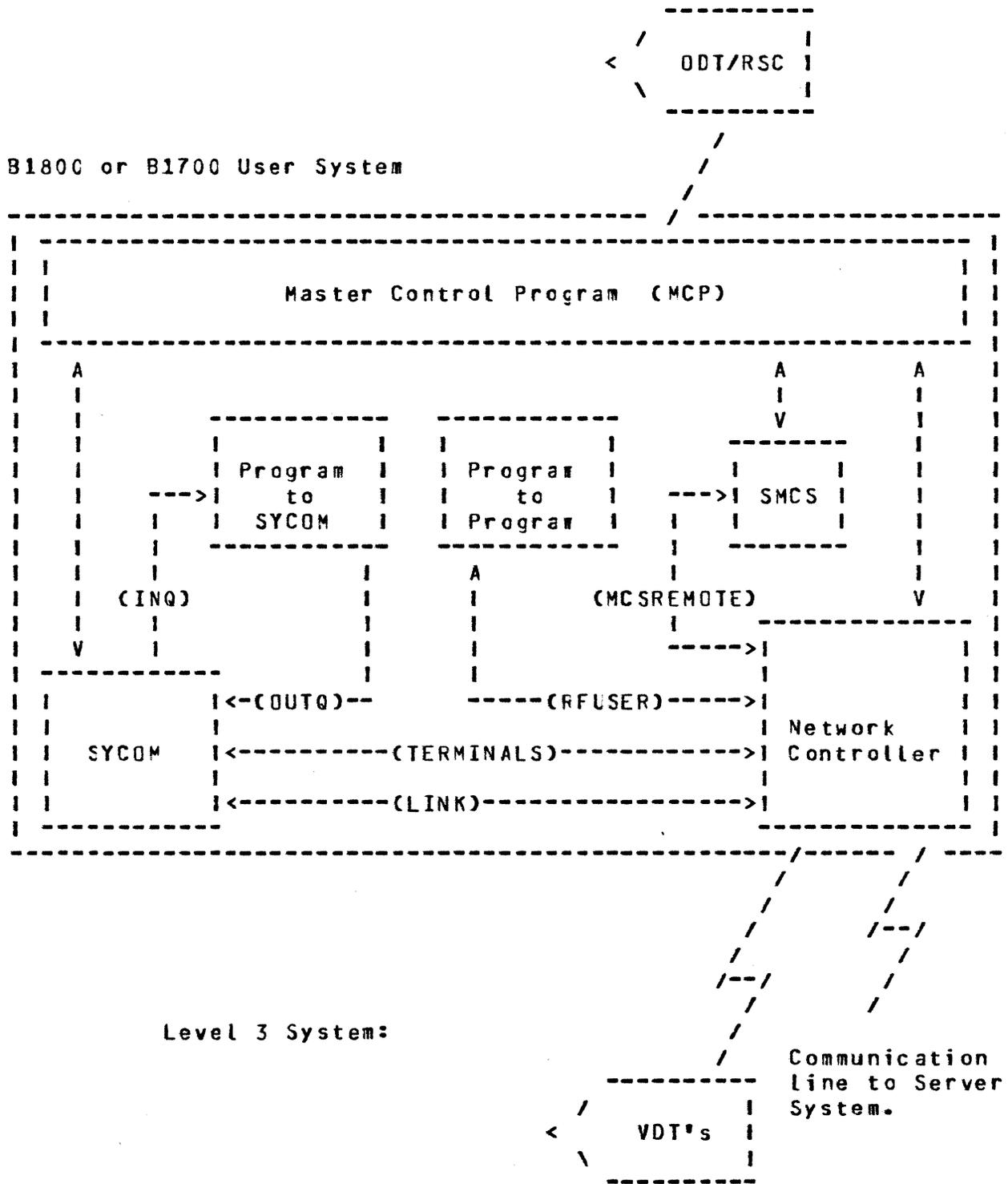


Figure 4-7: User System Schematic (SMCS; terminals)

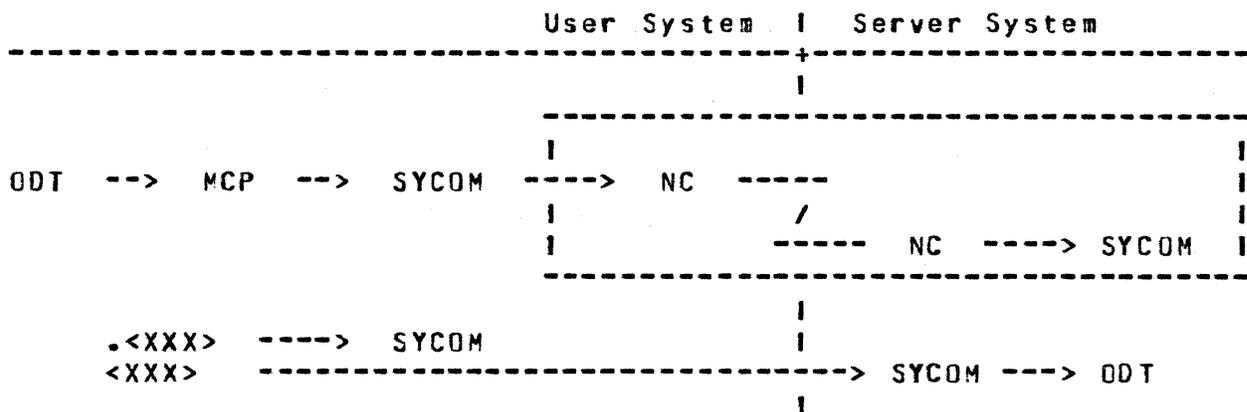
MESSAGE PATH ASSIGNMENTS

In order to describe the potential message communication paths opened up by the virtual terminal capability, a special message path syntax was developed to pictorially present this myriad of possibilities. The message path syntax is easy to understand as one can visualize these logical paths thru the User and Server Systems. System software elements are easily recognized by the widely used acronyms (all defined in the "Glossary"). Although the directional flow arrows "-->" indicate a single direction, communications are two-way. The boxed in syntactical elements to the right indicate an alternate message path; also available thru the "virtual terminal" capability.

FOR THE LEVEL 1 SYSTEM

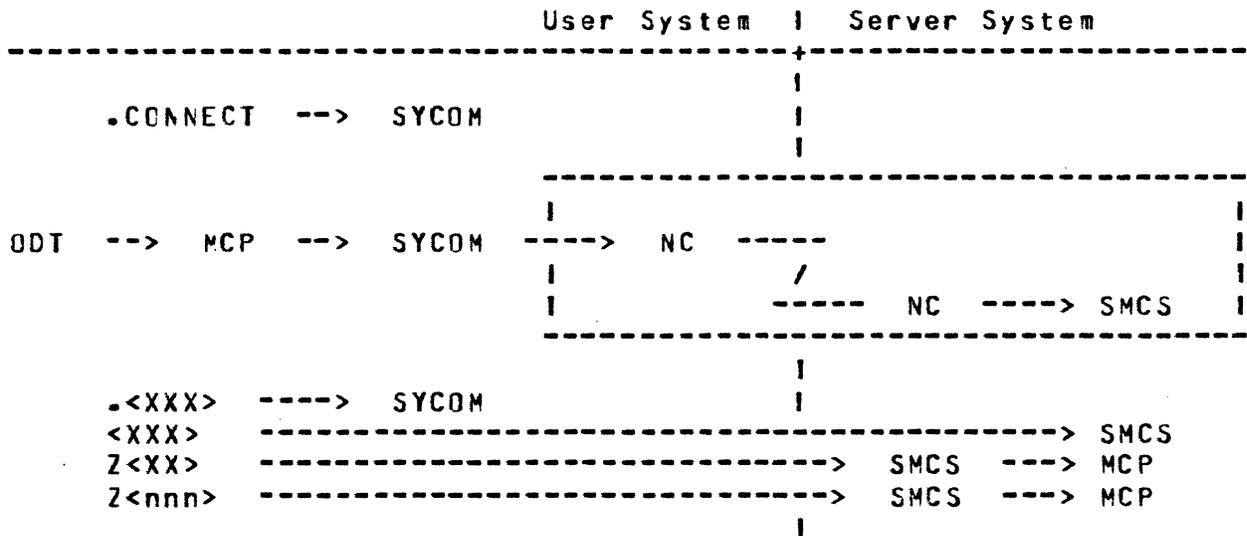
ODT input:

For this system level (Figure 3-9) there are three message paths for ODT input. The following examples explore each of these possibilities:

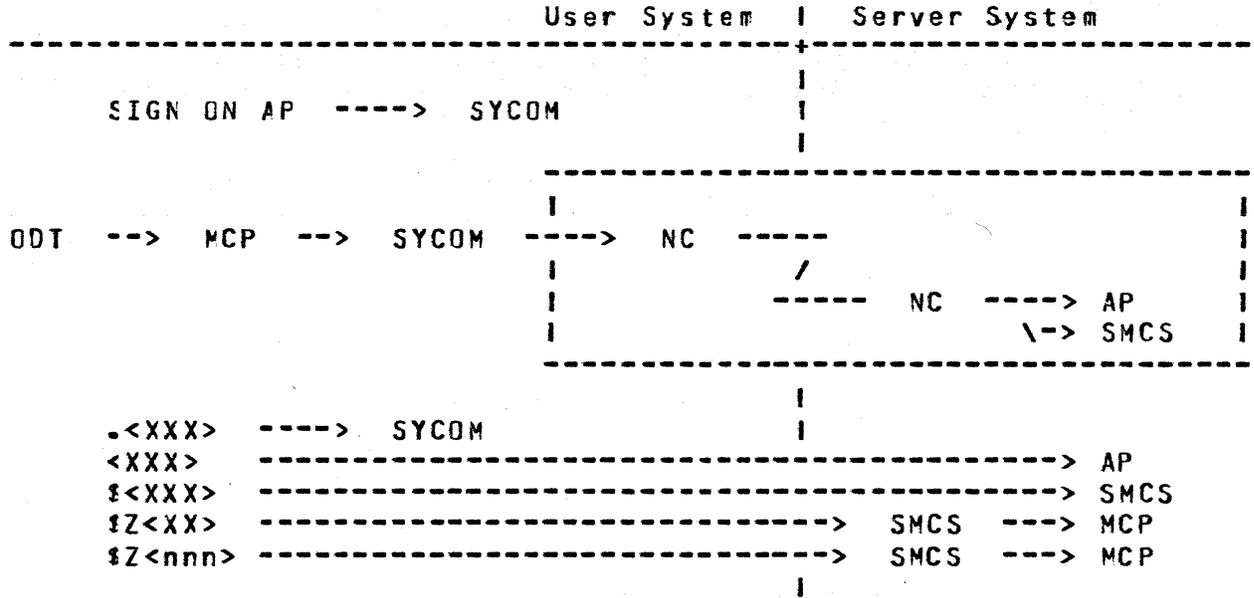


As shown in this diagram, all input command strings are given to User SYCOM by the MCP. Messages prefixed with a period "." are processed by User SYCOM. (This pseudo-signal character "." can be changed by using the "SIGNAL" command in SYCOM. It reverts back to the default ".", or whatever was specified by the DEFAULTPSEUDOSIGNAL parameter, when the SYCOM user signs-off.) All other input messages are sent to the Server System's SYCOM which will display them on that ODT.

When the "virtual terminal" capability is invoked (by using the ".CONNECT" command), the following message paths are established:

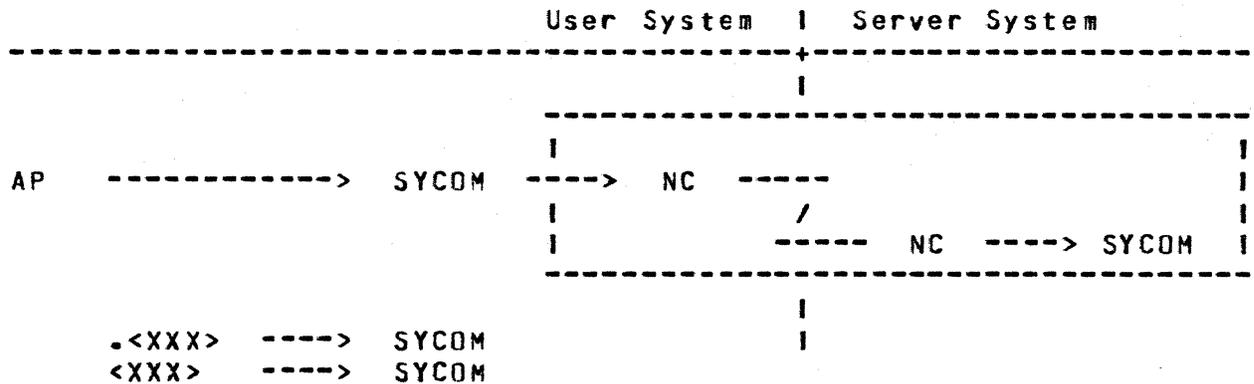


The SYCOM pseudo-signal character must immediately follow the AX without intervening blanks. Any number of blanks (including zero) may, however, follow the SYCOM pseudo-signal character. When a SMCS is running in the Server System, the ODT operator can either "sign-on" to a program in that system or initiate RSC type commands, thru the SMCS. When the virtual terminal is signed-on to the specified application program, it is "attached" to that program. All further non-prefixed input is sent to that program for processing. (In the example below, the "\$" character is assumed to be the Server System's SMCS "signal character"; it can be another character.)



Program-to-SYCOM input:

For "Program-to-SYCOM" communications, the following message path assignments are available:

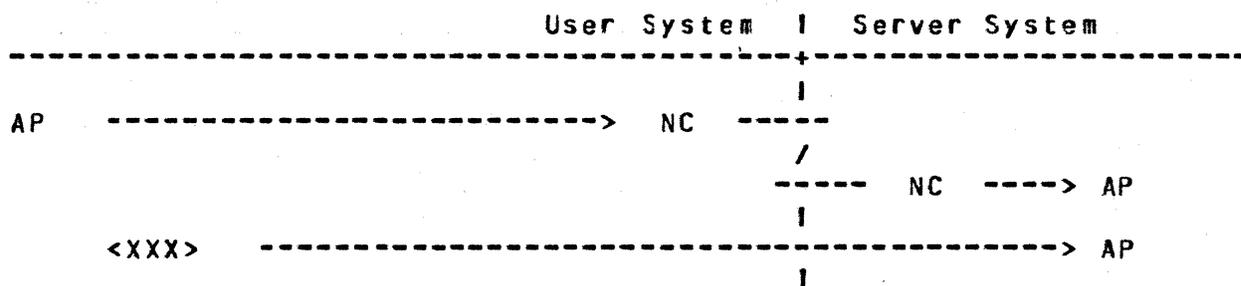


All input command strings are given to User SYCOM by the MCP via the "OUTQ" queue. Messages that are prefixed with a period "." are processed by User SYCOM.

When the "virtual terminal" capability is invoked (by entry of a ".CONNECT" command), the following message paths are established:

Program-to-program input:

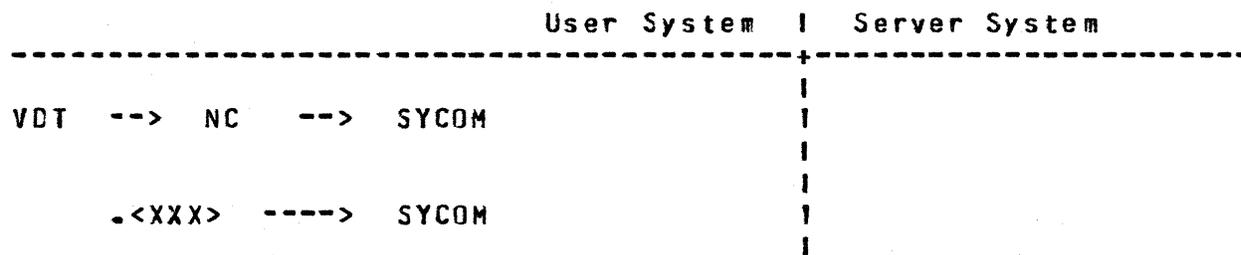
The message path for program-to-program communications is limited to the following predetermined path. Messages can be any format and contain any text agreed upon by both application programs. The only prevailing restrictions are maximum "buffersize" and all messages must be prefixed with a "header" ("Appendix E").



FOR THE LEVEL 2 SYSTEM

In this configuration (Figure 3-6), the ODT and all terminals attached to, and owned by, SYCOM normally communicate with local SYCOM. When the virtual terminal capability is utilized, these devices are permitted to communicate with either the system software programs or appropriate application programs in a Server System.

The ODT, program-to-SYCOM, and program-to-program input message paths for this software configuration are the same as those defined for the "Level 1 System". The following message path assignments are for local terminals attached to this system:

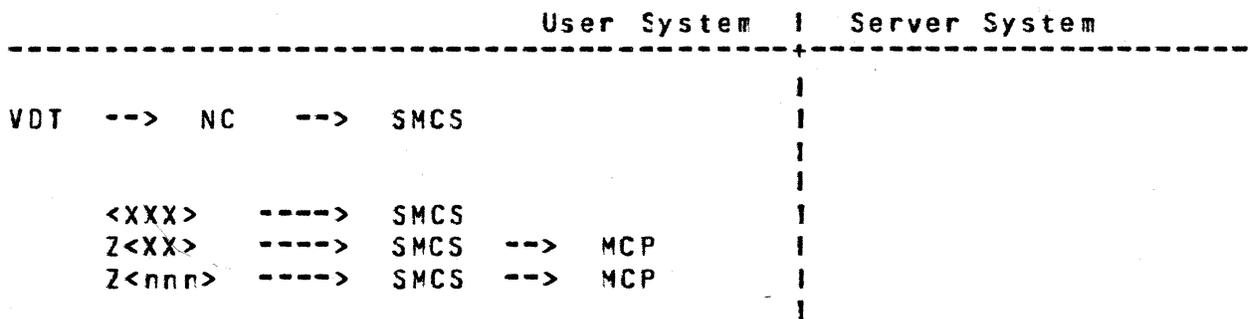


By entry of the ".CONNECT" command to SYCOM the virtual terminal capability is invoked. The following message paths are then established for the terminal:

FOR THE LEVEL 3 SYSTEM

In this configuration (Figure 3-7), all local terminals are attached to, and owned by the SMCS. Therefore, the normal message communication path is to the User System's SMCS. When the virtual terminal capability is utilized, these terminals are permitted to communicate with either system software programs or appropriate application programs in the Server System. Additionally, these same terminals are permitted to communicate with any appropriate application program in the User System.

The ODT, Program-to-SYCOM, and Program-to-Program input message paths for this software configuration are the same as those defined for the "Level 1 System". The following message path assignments are for all local terminals attached to the User System's SMCS. First the normal message paths for terminal input within the User System are presented:



In order for the terminal user to gain access to a SYCOM system, one of two access methods must be used: 1) By using the below SMCS "SIGN ON" command procedures, or 2) By utilizing the method whereby each command is "PASS'd" to SYCOM via the SMCS; it is up to the user. (Although this example is for a sign-on to SYCOM, it could be any appropriate application program in the User System.)



The following message path assignments are then in effect for all terminal input.

QUEUE DEPTH CONTROL

In the SYCOM operational environment, there is a MCP controlled data path which exists between SYCOM and the Network Controller. This data path is referred to as the "remote file" (or message queue) mechanism. Due to differences in data communication line transmission rates and peripheral input speeds, this remote file becomes a buffer. Without some kind of control on the message traffic thru this file, it could grow beyond reasonable limits or be jammed full.

To provide a control over the number of message blocks that can be queued up in this file at any given time, a "marker message" system is employed. As SYCOM is putting message blocks into this file, a count is maintained. When this count reaches the halfway point of the queue depth limit, a marker message is put into the file. The marker message is a specially prepared data message with a code in the message header. This special code ensures recognition by SYCOM when this message is echoed back to from the Network Controller.

Once a marker message is placed into the file, SYCOM continues putting message blocks into the file until the predetermined queue depth limit is reached, at which time peripheral input operations are suspended. When the Network Controller encounters the marker message during the transmission of the queued message blocks, it acknowledges receipt of a marker message by returning a "flag" in the corresponding "ACK" message to SYCOM. When SYCOM receives this acknowledgement, it resumes peripheral input, again places a marker message into the file and continues putting message blocks into the file until the queue limit is reached or until an end-of-file condition, whichever occurs first.

The current values of the "queue-size", and its associated marker-offset, may be queried or changed with the "QS" command during execution, or with the "QUEUESIZE" parameter during startup.

SYSTEM FILES

In the total system picture, there are many files and file types. However, as this system overview pertains only to SYCOM and its relationships, only those pertinent files which are required for SYCOM operations are presented. Following are file descriptions, by software element, that are necessary for SYCOM operations:

MASTER CONTROL PROGRAM (MCP) FILES

(SYSTEM)/USERCODE File

This file contains all valid usercode-passwords that are known to the system. It is used by the MCP to enforce file security disk files and programs/users that access them. With this file, the MCP can service any requests by programs regarding validity of specific usercode-password combinations. (This file is only required when the system is operating with "usercodes".)

NETWORK CONTROLLER FILES

Network Information (NIF) File

This file contains tables that describe the physical and logical attributes of the network. The initialization values for lines, terminals, and file tables of Network Controller are contained in this file. This file is created by the NDL compiler when it is in the process of generating a Network Controller.

SYCOM FILES

The Systems Communications Module utilizes many types of files in the management of its domain. There are, peripheral (includes disk), remote communications, and queue files. These files are briefly described here, under separate headings, with detail descriptions for major files available in the appendices.

Peripheral Files

These files are generally used to input/output RJE type data and include the card reader, card punch, line printer, magnetic tape, and disk files. Also there is a disk file which is used by SYCOM to manage the various options and settings. Following are general descriptions of the SYCOM files which fall into this category:

A. PARAMETERS File

The PARAMETERS file is a single record file containing required operational parameters for SYCOM. It is static and can only be changed by removal and recreation. This file is created by SYCOM upon the entry of appropriate parameters from either disk, card reader, or User ODT.

B. NEWPARAMS File

The "SYCOM/NEWPARAMS" file, on disk or cards, can be used to create the "SYCOM/PARAMETERS" file in preference to entering the "parameters" manually from the ODT. This file can be created using CANDE or any other supported text/editor program and is free-format; explanatory comments may or may not be included in the file.

C. Card Input File

This file is used to input RJE card data files from the system's card reader. It can be either a 80-column or 96-column device or a diskfile. However, in the case of the 96-column card reader, or a diskfile only the first 90 columns are processed.

D. Card Output File

This file is used to punch RJE card output data files to the system's card punch. It can be either a 80-column or 96-column device, as appropriate.

E. Printer File (optional)

This file is used to print all RJE print files that are being, or have been, received from a Server System. The line printer device can be any one of the various models that can be attached to the system; which can be either a 120-position or 132-position line printer. If no printer is available, the printer-backup files may be transmitted to a "remote system" which has a printer.

There is an additional print file which is used to list SYCOM's "TRACE" or "OLDTRACE" files via the ".LIST" command; it does not interfere with printing the RJE type print files received by SYCOM.

Remote Files

The remote files are used in communications between SYCOM and the Network Controller and SYCOM and either the terminals or SMCS. There are two of these files which are described as follows:

A. LINK File

The "LINK" file is used for all input/output messages relating to RJE and file transfers; specifically, station types for Control, ODT (SPO), Cards, Printer, and File records.

E. TERMINALS File

The "TERMINALS" file is the communication link between SYCOM and the Network Controller for local terminal messages. This remote file is however, only used when there are local terminals; controlled by either the SMCS or SYCOM.

Queue Files

The queue files are utilized by SYCOM for buffered communications between itself, the MCP, and the Network Controller. The purpose of each of these queue's is described as follows:

A. RJE Queue

Used to save printer/punch data received from the other system when the User System's printer or punch is busy.

E. LINK Queue

1. Used to save input messages from the other system during the "initialization" phase; else they may be lost.
2. Used to save output messages for the other system when "messages queued" exceeds "MAX-MESSAGES".

C. INQ Queue

This queue is the mechanism by which SYCOM sends reply messages to an application program dealing in program-to-SYCOM communications. SYCOM "writes" into it and the application program "reads" from it to obtain messages.

E. QUIQ Queue

This queue is the mechanism whereby SYCOM receives input command messages from an application program that deals in program-to-SYCOM communications. The application program "writes" into it and SYCOM "reads" from it to obtain input messages.

RESTRICTIONS

The following restrictions are valid for the system based on a logical evaluation of the system's requirements:

- Maximum message block size = 1950 bytes
- Minimum message block size = 362 bytes

- Maximum simultaneous system connections = 1
- Maximum message block queue depth = 255
- Maximum number of file transfers queued to process = 99
- Maximum number of terminals supported = 15
- Maximum number of card readers supported = 1
- Maximum number of card punches supported = 1
- Maximum number of line printers supported = 1
- Neither LCAD.DUMP or SYSTEM/COPY tapes are supported for file transfer operations.
- Only "data type files" are file transferable between disk/tape and tape/disk (codes files are invalid). (Only one data file per tape is allowed.)
- 9.0 type files, i.e., type > 15, are not supported for file transfers.
- Multi-point networks are not supported by SYCOM; at this time.
- Message blocking is prohibited. (A message block already contains multiple records or data images.)
- Monitoring and controlling of programs in the Server System requires a MCS executing in that system.
- This system does not support "binary-synchronous" communications.

These restrictions may be changed during subsequent development cycles, but are however, currently in effect.

OPERATOR COMMANDS

GENERAL

The set of commands described in the section is used to direct the System Communications Module (SYCOM) program operations from either the Operator Display Terminal (ODT), a remote operator display terminal (remote ODT), a system terminal, or an application program running in the system.

Non-SYCOM program commands are also permitted to direct user-system and server-system operations by way of respective SMCS's.

A remote ODT station is a station that is known to SMCS as a remote ODT station and is defined in the network controller with a transmit frequency greater than 249.

A workstation is a station that is known to SMCS as a workstation or controller station and is defined in the network controller with a transmit frequency less than or equal to 249.

Table 4-1 at the end of this section contains a summary of all the SYCOM commands available in the SYCOM program.

INPUT COMMANDS

Each statement in the Command Language consists of a command and, in some cases, a list of parameters. The parameters in the list are separated from the command and each other by blanks. In the following examples single spaces are used although the use of multiple spaces between command elements could be used.

When the SYCOM has finished processing a command, it displays a reply message either confirming successful completion of the command or indicating an error. Errors terminate the processing of a command; the entire command must be re-entered correctly (except for commands which allow multiple operands such as .LIST).

presentation. To provide a clear line of continuation for the command string, the use of line direction arrows are used. A line direction arrow appears as ----> and always flows left to right; to continue the command string. It is easy to follow a command string in this format as it is logical in nature.

OPEN-CLOSE PARENTHESES ()::

When these characters appear in a command syntax statement they are required input. See the "COPY" command for a use reference.

SLASH /:

When this character appears in a command syntax statement it is required input. A space or spaces on either side of this character are not allowed.

END OF STATEMENT:

The end of a command statement is indicated by the appearance of a % (percent sign). It is a syntactical string terminator but is only required if commentary text is included in the command input.

Punctuation Rules

The space, the comma, and the period are considered punctuation marks (except in literals) and are used as follows:

1. The space (blank) is a separator, required after words and literals in the absence of any other separator. A space may precede or follow any other separator, and many spaces are the same as one (except in literals).
2. A comma is a separator that is legal only where it is specifically indicated in the command syntax. The comma, where it is legal, can also serve as a terminator for words and numeric literals. The comma is never required, except to separate criteria in a command string.
3. The period is a required prefix to local SYCOM commands. It is never required within any command string syntax. Its only acceptability would be as an embedded character in a file name, system identifier, or usercode/password.

Comments

1. SYCOM commands and/or operands may be entered in either upper or lower case. File-names, however, must appear exactly as recorded in the system disk directory. No translation will occur. Currently any quote marks that

are encountered in the command string will be taken only to mean a quote mark, not to start a quoted string.

2. Passwords containing either the special character "," or ")" will not be recognized.

COMMAND RELATED ERROR MESSAGES

Error messages specific to the following SYCOM commands appear in Appendix A (Command Error Messages).

ABORT

The ABORT command causes the SYCOM program to abort an in-process file transfer or a scheduled file transfer.

Syntax:

```
.ABORT----- CONNECT ----->
      |-- FETCH --|      |-- <schedule number> --|
      |-- PUT ----|      |-- = -----|
>-----|
      |-- % <comments> --|
```

Semantics:

CONNECT	Causes a scheduled connect to be removed from the schedule. Connects will only be scheduled when the remote system is off line.
FETCH	Causes the in-process file transfer initiated by the FETCH command to be aborted. The file being received is not saved.
PUT	Causes the in-process file transfer initiated by the PUT command to be aborted.
schedule number	Identifies the file in the file transfer schedule to be aborted. If the field is left blank, the FETCH or PUT operation in process is aborted (if any).
=	Aborts the in-process file transfer and all files scheduled to be transferred. The equal sign (=) character must follow PUT or FETCH. ABORT FETCH = or ABORT PUT = is only valid from the system CDT.

Examples:

- .ABORT FETCH % Aborts the file transfer in process which was
 % initiated by a FETCH command.

- .ABORT PUT % Aborts the file transfer in process which was
 % initiated by a PUT command.

- .ABORT FETCH 4 % Aborts the file transfer, schedule number 4
 % in the PUTQUEUE of the remote system, whether
 % in process or not.

- .ABORT PUT 10 % Aborts the file transfer, schedule number 10
 % in the PUTQUEUE of the local system, whether
 % in process or not.

- .ABORT FETCH = % Aborts all file transfers in the PUTQUEUE of
 % the remote system and the in process file
 % transfer (if any) which was initiated by a
 % FETCH command.

- .ABORT PUT = % Aborts all file transfers in the PUTQUEUE of
 % the local system and the in process file
 % transfer (if any) which was initiated by a
 % PUT command.

The SYCOM message PUT REQUESTOR MISMATCH can occur if the user attempts to abort a PUT or FETCH operation of a file and the source file was secured with a usercode/password.

Any one of the following conditions must exist to initiate a file transfer of a secured file.

1. The user must be logged on to the SMCS with a usercode/password and enter the PUT or FETCH command to the SYCOM program through the PASS command to the SMCS.

Examples:

```
PASS SYCOM PUT <filename>  
PASS SYCOM FETCH <filename>
```

2. The user must be logged-on to the SYCOM program by entering the US command to the program.

Example:

```
US = USER/JONES
```

3. The user must assign a usercode/password to the source file when a COPY, PUT, or FETCH SYCOM command is entered to the SYCOM program.

BURROLGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

5-7
COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
P. S. 2219 0458 (A)

Examples:

- .PUT MASTER (US = DOE/JOHN)
- .FETCH FILEA (US = USER/PAUL)
- .COPY SOURCE (US = ODT) FROM (HOST = B1800)

is aborted.

- BINARY** Applies to data files and causes the SYCOM program to expand all the records in a file prior to transmission. Code files are always expanded. The BINARY option allows users to transfer files to and from an ASCII system. If the BINARY option is not specified for a data file transfer to a non-EBCDIC system, the SYCOM program changes non-graphic characters to question mark (?) characters. The graphic character set is the EBCDIC representation of the 95 valid ASCII characters and the null character.
- If a data file is sent to a CMS system (ASCII machine) with the BINARY option specified, to retrieve the same file and maintain file integrity, the ADD command must include the BINARY option.
- FILE** Allows SYCOM to read the names of the file(s) to be transmitted from the file, <filename-1>. By using the FILE option, a series of files can be transmitted by one ADD statement. Each filename must be entered on separate records in the file. If the destination filename is to be different than the source filename, it is entered following the source filename separated by at least one blank. Comments can follow the last filename on the record starting with a percent sign ("%").
- filename-1** Specifies the name of the file to be transmitted. The filename can be a single file or a family of files in the format: "<multifile-id>"/=.
- filename-2** Assigns the name to be given to the file on the destination system. The filename can be a single file or a family of files in the format: "<multifile-id>"/=.
- KIND** KIND is a keyword which is used to describe the peripheral device of a file. The KIND keyword is not required if the file is on disk.
- LIST** Informs the receiving system that the printer backup file is to be printed after transmission by zipping a command to the SYSTEM/BACKUP program if the MCP autobackup option is set.

- NO Overrides the default LIST attribute and is only valid for printer backup files.
- REMOVE Causes the file to be removed from the sending system after the transfer has been completed. REMOVE is the default for printer backup files.
- SAVE Ensures that a printer backup file is saved after the file is transferred on the sending system. SAVE is the default for non-printer backup file transfer.
- SKIP <n> Specifies how many files to skip if the "ADD <multifile-id>/=" or the indirect (FILE) option is used. The first "<n>" files of the family are skipped. Usage of the keyword is optional.
- system-id-1 Specifies the system identification of the system from which the file is to be transferred from.
- system-id-2 Specifies the system identification of the system that is to receive the file. Identifies the user for the purpose of verifying access rights to the specified filename.
- volume-1 Specifies the pack-id or tape-id for the source file. This option is not required for files on the system disk, or if a pack-id appears in <filename-1>. "DISK" is used to indicate the system disk.
- volume-2 Specifies the pack-id or tape-id for the destination file. This option is not required for files on the system disk, or if a pack-id appears in <filename-2>.
- () Parentheses are key characters when using a US, NO LIST, REMOVE, or SAVE option. The options can appear in any order within the parentheses.
- * Assigns the filename specified in filename-1 to filename-2.

Examples:

```
.ADD XYZ (US=US/PASS) FROM (DISK, HOST=USER1) TO (DISK, HOST=SERVER1)
.ADD DMPALL AS DUMPALL FROM (HOST=FLB, KIND=TAPE)
.ADD SDL/= AS SDLNEW/= FROM (HOST=HUB)
.ADD RELEASE (FILE, NO LIST) TO (HOST=HUB)
.ADD = (US A/B, SKIP 123) TO (HOST=USER3)
.ADD RELEASE (FILE, SKIP 9) TO (HOST=USER5)
```

Examples one and two are examples of single file transfer. Example three is an example of a transfer of a family of files and example four is an example of using a list of files to effect a transfer. Example five is an example of using the SKIP option when transmitting a family of files. Example six is an example of using a list of files with the SKIP option.

BUFFER

The BUFFER command causes the SYCOM program to change or interrogate the buffer size for sending file transfers. The buffer size is specified in bytes. The minimum buffer size is 362 bytes and the maximum is 1950 bytes. If the BUFFER parameter is not specified in the parameters file, the buffer size defaults to 1950 bytes.

The receiving buffer size is always 1950 bytes.

If the buffer size is changed with the BUFFER command, the new value is effective until another BUFFER command is entered or the SYCOM program goes to end-of-job. The buffer size for the next execution of the SYCOM program reverts back to the value specified by the BUFFER parameter in the SYCOM parameters file.

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the value of this parameter.

Syntax:

```
.BUFFER -----  
|-- = --| |-- <integer> --| |-- % <comments> --|
```

Semantics:

integer Specifies the number of bytes for the buffer used in file transfers. The value must be between 362 and 1950, inclusive.

```
.BUFFER = 400  
.BUFFER 1950  
.BUFFER % Queries the current transmission buffer size.  
% The receiving buffer size can also be  
% interrogated by entering the LIST PARAMETERS  
% command to the SYCOM program.
```

BYE

The BYE command causes the SYCOM program to log-off the current user from the SYCOM program. It is valid from all states including the system ODT and application programs. If the user is connected to the server-system SMCS by way of the COM command, an automatic DISCONNECT is initiated. If the terminal is not owned by the SYCOM program, a DETACH message is sent to the local SMCS. Also, the pseudo-signal character is reset to the default pseudo-signal character.

Syntax:

```
.BYE -----  
      |-- % <comments> --|
```

Example:

```
.BYE
```

CL (Clear Device)

The CL command causes the SYCOM program to clear a peripheral device if the device becomes inoperable while receiving a data file. The buffers associated with the specified device are cleared and the receiving process is terminated.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.CL ----- LP -----  
|--- CP ---| |-- % <comments> --|
```

Semantics:

LP Clears the line printer.

CP Clears the card punch.

Examples:

```
.CL LP  
.CL CP
```

CONNECT

The CONNECT command causes the SYCOM program to log-on a terminal user to the server-system SMCS.

The SYCOM program ensures that a MCS-type program is running in the server-system and that a virtual station slot is available before making the connection.

If a user is logged-on to the SYCOM program, then that user's usercode/password is forwarded to the MCS in the other system which automatically logs the user on to that MCS.

If switched data communication lines are being used, an Automatic Calling Unit (ACU) is present, and a telephone number is provided, then the SYCOM program initiates the dial-out.

Syntax:

```
.CONNECT -----  
      |-- % <comments> --|
```

Example:

```
.CONNECT
```


Notifies the server-system to create a new file with the name specified as filename-2. If the file is present on the server-system, the command is aborted.

BINARY

Applies to data files and causes the SYCOM program to expand all the records in a file prior to transmission. Code files are always expanded. The BINARY option allows users to transfer files to and from an ASCII system. If the BINARY option is not specified for a data file transfer to a non-EBCDIC system, the SYCOM program changes non-graphic characters to question mark (?) characters. The graphic character set is the EBCDIC representation of the 95 valid ASCII characters and the null character.

If a data file is sent to a CMS system (ASCII machine) with the BINARY option specified, to retrieve the same file and maintain file integrity, the COPY or FETCH command must include the BINARY option.

FILE

Allows SYCOM to read the names of the file(s) to be transmitted from the file, <filename-1>. By using the FILE option, a series of files can be transmitted by one COPY statement. Each filename must be entered on separate records in the file. If the destination filename is to be different than the source filename, it is entered following the source filename separated by at least one blank. Comments can follow the last filename on the record starting with a percent sign ("%").

filename-1

Specifies the name of the file to be transmitted. The filename can be a single file or a family of files in the format: "<multifile-id>"/=.

filename-2

Assigns the name to be given to the file on the destination system. The filename can be a single file or a family of files in the format: "<multifile-id>"/=.

KIND

KIND is a keyword which is used to describe the peripheral device of a file. The KIND keyword is not required if the file is on disk.

LIST

to be printed after transmission by zipping a command to the SYSTEM/BACKUP program if the MCP autobackup option is set.

NO

Overrides the default LIST attribute and is only valid for printer backup files.

REMOVE

Causes the file to be removed from the sending system after the transfer has been completed. REMOVE is the default for printer backup files.

SAVE

Ensures that a printer backup file is saved after the file is transferred on the sending system. SAVE is the default for non-printer backup file transfer.

SKIP <n>

Specifies how many files to skip if the "COPY <multifile-id>/=" or the indirect (FILE) option is used. The first "<n>" files of the family are skipped. Usage of the keyword is optional.

system-id-1

Specifies the system identification of the system from which the file is to be transferred from.

system-id-2

Specifies the system identification of the system that is to receive the file.

usercode and password

Identifies the user for the purpose of verifying access rights to the specified filename.

volume-1

Specifies the pack-id or tape-id for the source file. This option is not required for files on the system disk, or if a pack-id appears in <filename-1>. "DISK" is used to indicate the system disk.

volume-2

Specifies the pack-id or tape-id for the destination file. This option is not required for files on the system disk, or if a pack-id appears in <filename-2>.

()

Parentheses are key characters when using a US, NO LIST, REMOVE, or SAVE option. The options can appear in any order within the parentheses.

Assigns the filename specified in filename-1 to filename-2.

Examples:

.COPY XYZ (US=US/PASS) FROM (DISK, HOST=USER1) TO (DISK, HOST=SERVER1)
.COPY DMPALL AS DUMPALL FROM (HOST=HUB, KIND=TAPE)
.COPY SDL/= AS SDLNEW/= FROM (HOST=HUB)
.COPY RELEASE (FILE, NO LIST) TO (HOST=HUB)
.COPY = (US A/B, SKIP 123) TO (HOST=USER3)
.COPY RELEASE (FILE, SKIP 9) TO (HOST=USER5)

Examples one and two are examples of single file transfer. Example three is an example of a transfer of a family of files and example four is an example of using a list of files to effect a transfer. Example five is an example of using the SKIP option when transmitting a family of files. Example six is an example of using a list of files with the SKIP option.

CP NR (Card Punch Not Ready)

The CP NR command causes the SYCOM program to inform the user-system that the system card punch is in a Not Ready state. The SYCOM program in turn informs the server-system of the condition of the hardware. The server-system suspends transmitting any card punch data stream messages currently in process.

It is the responsibility of the user-system operator to inform SYCOM when the card punch is Ready to receive output.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.CP NR -----]
      |-- % <comments> --|
```

Example:

```
.CP NR
```

DEBUG

The DEBUG command causes the SYCOM program to print all data messages read or written by the SYCOM program. If no option is specified, a request for the current status of the DEBUG facility is displayed.

This parameter can only be interrogated by a workstation. A remote ODI or the system ODI can change the setting of this parameter.

Syntax:

```
.DEBUG -----|  
      |-- OFF  |--|  |-- % <comments>  |--|  
      |-- ON   |--|  
      |-- DUMP |--|
```

Semantics:

OFF Terminates the debugging option.

ON Starts the debugging option.

DUMP Causes a dump and list trace of the SYCOM program to be initiated. Also, a similar request is sent to the remote SYCOM program.

Example:

```
.DEBUG  
.DEBUG ON  
.DEBUG OFF  
.DEBUG DUMP
```

DISCONNECT

The DISCONNECT command causes the SYCOM program to terminate a session with a host server-system. The terminal is logged-off the server-system when a DISCONNECT message is sent to the other system. The terminal reverts back to its previous state of being signed on to the SYCOM program. The terminal user is then free to enter more commands to the SYCOM program.

Syntax:

```
.DISCONNECT -----  
                |-- Z <comments> --|
```

Example:

```
.DISCONNECT
```

The pseudo-signal character must be included in the DISCONNECT command; otherwise, if the terminal is connected to the remote system, the DISCONNECT command is sent directly to the remote system SMCS and the following message is displayed or it can be sent to the application program that the user was signed on to.

```
>> ERROR: MCP REJECTED ZIP CONTROL STRING  
INVALID MNEMONIC-DIS
```

The current pseudo-signal character is not changed by the DISCONNECT command. Only the BYE or SIGNAL SYCOM commands or the *OFF SMCS command causes the pseudo-signal character to be reset to the default pseudo-signal character.

EOJ

The EOI command causes an orderly shutdown of the SYCOM program. No commands are processed after the EOI command is entered. The message PLEASE SIGN OFF is broadcast to all terminals currently logged-on to the SYCOM program. A close message is sent to the other system's SYCOM program after the completion of any in-process file transfers and when there are no longer any terminals connected to the remote system. The SYCOM program then sends a disconnect message to the network controller and goes to end-of-job.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.EOJ -----  
      |-- [IT] |--|      |-- % <comments> |--|  
      |-- xT   |--|
```

Semantics:

D causes the SYCOM program to initiate a dump.

DT Causes the SYCOM program to initiate a dump and a list trace.

xT Causes the SYCOM program to initiate a list trace. The "x" character represents any non-blank character except the "D" character.

Examples:

```
.EOJ  
.EOJ D % Causes a dump.  
.EOJ DT % Causes a dump and list trace.  
.EOJ xT % Causes a list trace. The "x" character represents  
% any non-blank character except the "D" character.
```


statement. Each filename must be entered on separate records in the file. If the destination filename is to be different than the source filename, it is entered following the source filename separated by at least one blank. Comments can follow the last filename on the record starting with a percent sign ("%").

filename-1

Specifies the name of the file to be transmitted. The filename can be a single file or a family of files in the format: "<multifile-id>"/=.

filename-2

Assigns the name to be given to the file on the destination system. The filename can be a single file or a family of files in the format: "<multifile-id>"/=.

KIND

KIND is a keyword which is used to describe the peripheral device of a file. The KIND keyword is not required if the file is on disk.

LIST

Informs the receiving system that the printer backup file is to be printed after transmission by zipping a command to the SYSTEM/BACKUP program if the MCP autobackup option is set.

NO

Overrides the default LIST attribute and is only valid for printer backup files.

REMOVE

Causes the file to be removed from the sending system after the transfer has been completed. REMOVE is the default for printer backup files.

SAVE

Ensures that a printer backup file is saved after the file is transferred on the sending system. SAVE is the default for non-printer backup file transfer.

SKIP <n>

Specifies how many files to skip if the "FETCH <multifile-id>/" or the indirect (FILE) option is used. The first "<n>" files of the family are skipped. Usage of the keyword is optional.

system-id-1

Specifies the system identification of the system from which the file is to be transferred from.

system-id-2

Specifies the system identification of the system that is to receive the file.

usercode and password

Identifies the user for the purpose of verifying access rights to the specified filename.

volume-1

Specifies the pack-id or tape-id for the source file. This option is not required for files on the system disk, or if a pack-id appears in <filename-1>. "DISK" is used to indicate the system disk.

volume-2

Specifies the pack-id or tape-id for the destination file. This option is not required for files on the system disk, or if a pack-id appears in <filename-2>.

()

Parentheses are key characters when using a US, NO LIST, REMOVE, or SAVE option. The options can appear in any order within the parentheses.

Assigns the filename specified in filename-1 to filename-2.

Examples:

```
.FETCH DMPALL
.FETCH DMPALL FROM USER1 TO USER2
.FETCH USER/COBOL/= TO DISK
.FETCH COBOL/= TO USER
.FETCH = (US A/B, SKIP 9)
.FETCH RELEASE (FILE, NO LIST)
.FETCH SYCOM AS HUB FROM PACKA TO PACKB % same as following example
.FETCH PACKA/SYCOM/ AS PACKB/HUB/ % same as preceding example
.FETCH SDL/= AS SDLNEW/= TO PACKC % transfer of a family of
files
.FETCH = (US A/B, SKIP 9) % all (A)/= except first 9
files
.FETCH PACKD/COBOL74/= TO DISK % "DISK" as volume id means
system disk
.FETCH COBOL74/= FROM PACKD % same as preceding example
```

The first example will use the same destination name as the source name. The second example uses different pack ids. The third example takes a family of files from a user pack to the system disk. The fourth example transfers a family of files from the system disk to the pack USER. The fifth example will fetch all the files under the usercode, "A", except the first nine. The sixth example is an example of using a list of files.

GROUPSIZE

The groupsize command displays or changes the groupsize used for dividing families of files into groups.

The groupsize is used to restrict the number of files that will be sent a one time when sending an entire family of files (i.e. PUT SYSTEM/=).

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the value of this parameter.

Syntax

```
.GROUPSIZE -----|  
                |           |  
                |----- <integer> -----|
```

Semantics:

integer One or two digit integer value between one and 99 that represents the new value of the groupsize.

Example:

```
.GROUPSIZE  
.GROUPSIZE 15
```

LIST

The LIST command displays information about the system and its users. Command options provide lists of those users currently signed on, terminals that are currently on-line, and file transfers that are currently scheduled to occur.

Syntax:

```
          |<-----|
          |
.LIST  ----- OLDTRACE -----|
          |--- PARAMETERS ---|      |-- % <comments> --|
          |--- PUTQUEUE ---|
          |--- SCHEDULE ---|
          |--- TERMINALS ---|
          |--- TRACE ---|
          |--- USERS ---|
```

Semantics:

OLDTRACE Prints the last trace file created by SYCOM.

PARAMETERS Displays the parameters SYCOM is using.

PUTQUEUE Prints the current range of file transfer requests that were initiated by a PUT request. The list provides a list of up to the last 99 PUT requests.

SCHEDULE Displays all file transfers currently in the schedule in schedule number order.

TERMINAL Displays all terminals that are currently on line.

TRACE Prints the current trace file being written by SYCOM.

USERS Displays all of the users of the SYCOM program.

Examples:

```
.LIST OLDTRACE
.LIST PARAMETERS
.LIST PUTQUEUE
.LIST SCHEDULE
.LIST TERMINALS
.LIST TRACE
.LIST USERS
.LIST OLDTRACE PARAMETERS PUTQUELE SCHEDULE TERMINALS TRACE USERS
```

LOG

The LOG command causes the SYCOM program to display the number of occurrences of specific data communication line errors and certain message transmission statistics. The totals in the counters can be initialized if the CLEAR option is included in the command.

Syntax:

```
.LOG -----  
      |-- CLEAR --|   |-- % <comments> --|
```

Semantics:

CLEAR Resets the log counters and closes the statistics file.

Example:

```
.LOG  
.LOG CLEAR
```

LP

The LP command informs the user-system SYCOM program that the line printer is Not Ready. The SYCOM program informs the server-system of this temporary condition. The server-system then suspends transmission of the line printer data stream messages until notified to resume by the user-system SYCOM program.

It is the responsibility of the user-system operator to inform SYCOM when the line printer is again Ready to receive output data.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.LP NR -----  
      |-- % <comments> --|
```

Example:

```
.LP NR
```

OFFLINE

The OFFLINE command causes the SYCOM program to break the existing connection with the other system, go into standby mode, and wait for the other system to call back. This command can be used for either a switched or non-switched Data Comm line connection. When a "DLE ENQ" is received from the other system, the network controller re-establishes the connection with the other system and normal operations resume.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.OFFLINE -----  
          |-- % <comments> --|
```

Example:

```
.OFFLINE
```

ONLINE

The ONLINE command causes the network controller to attempt to re-establish the line connection to the server-system.

The error message RETRIES-UP is displayed when the current message buffer being sent to the server-system is not being received. By entering the ONLINE command, the message buffer being sent is discarded and an attempt is made to re-establish the line connection.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.ONLINE -----|  
      |-- % <comments> --|
```

Example:

```
.ONLINE
```


Examples:

.PHONE 1-222-333-4444	% Assigns a new telephone number.
.PHONE =	% Clears any prior telephone number.
.PHONE	% Queries the current telephone number.

PROGRAM

The PROGRAM command is used to interrogate, enable, or disable program input to SYCOM.

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the setting of this parameter.

Syntax:

```
.PROGRAM -----!  
          |                |  
          |----- ON ----|  
          |                |  
          |----- OFF ----|
```

Semantics:

ON Turns the program input parameter on.
OFF Turns the program input parameter off.

Examples:

```
.PROGRAM  
.PROGRAM ON  
.PROGRAM OFF
```


records in the file. If the destination filename is to be different than the source filename, it is entered following the source filename separated by at least one blank. Comments can follow the last filename on the record starting with a percent sign ("%").

filename-1

Specifies the name of the file to be transmitted. The filename can be a single file or a family of files in the format: "<multifile-id>/=".

filename-2

Assigns the name to be given to the file on the destination system. The filename can be a single file or a family of files in the format: "<multifile-id>/=".

KIND

KIND is a keyword which is used to describe the peripheral device of a file. The KIND keyword is not required if the file is on disk.

LIST

Informs the receiving system that the printer backup file is to be printed after transmission by zipping a command to the SYSTEM/BACKUP program if the MCP autobackup option is set.

NO

Overrides the default LIST attribute and is only valid for printer backup files.

REMOVE

Causes the file to be removed from the sending system after the transfer has been completed. REMOVE is the default for printer backup files.

SAVE

Ensures that a printer backup file is saved after the file is transferred on the sending system. SAVE is the default for non-printer backup file transfer.

SKIP <n>

Specifies how many files to skip if the "PUT <multifile-id>/=" or the indirect (FILE) option is used. The first "<n>" files of the family are skipped. Usage of the keyword is optional.

system-id-1

Specifies the system identification of the system from which the file is to be transferred from.

system-id-2

Specifies the system identification of the system that is to receive the file.

usercode and password

Identifies the user for the purpose of verifying access rights to the specified filename.

volume-1

Specifies the pack-id or tape-id for the source file. This option is not required for files on the system disk, or if a pack-id appears in <filename-1>. "DISK" is used to indicate the system disk.

volume-2

Specifies the pack-id or tape-id for the destination file. This option is not required for files on the system disk, or if a pack-id appears in <filename-2>.

()

Parentheses are key characters when using a US, NO LIST, REMOVE, or SAVE option. The options can appear in any order within the parentheses.

Assigns the filename specified in filename-1 to filename-2.

Examples:

```
.PUT DMPALL
.PUT DMPALL FROM USER1 TO USER2
.PUT USER/COBOL/= TO DISK
.PUT COBOL/= TO USER
.PUT = (US A/B, SKIP 9)
.PUT RELEASE (FILE, NO LIST)
.PUT RELFILES (FILE, NO LIST, SAVE)
  % file RELFILES contains list of files to be sent, with
  % destination names if different from source names. NO
  % LIST will prevent automatic AP at destination end if
  % these files are required to be on disk for a subsequent
  % SYSTEM/COPY to release tape.
```

The first example will use the same destination name as the source name. The second example uses different pack ids. The third example takes a family of files from a user pack to the system disk. The fourth example transfers a family of files from the system disk to the pack USER. The fifth example will put all the files under the usercode, "A", except the first nine. The sixth example is an example of using a list of files.

QS (Queue Size)

The QS command causes the SYCOM program to change or interrogate the QUEUESIZE parameter. If the QS command is not followed by an equal sign or number, the current value is displayed. If the QS command is followed by a number between 4 and 15, inclusive, the SYCOM program changes the QUEUESIZE parameter accordingly.

If the QUEUESIZE parameter has been changed by entering the QS command and the SYCOM program goes to end-of-job, the value of QUEUESIZE parameter reverts back to the value specified in the parameters file.

The equal sign (=) character is optional for the QS command.

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the value of this parameter.

Syntax:

```
.QS -----|  
  |-- = --|  |-- <integer> --|  |-- % <comments> --|
```

Semantics:

= Optional key character which can be used when changing the value of QUEUESIZE.

integer One or two digit integer value between 4 and 15 which assigns the new queue depth limit.

Examples:

```
.QS=9  
.QS 12  
.QS % Displays current value of the queuesize and  
% associated marker offset.
```

READ

The READ command causes the SYCOM program to open a card-image file and begin transmitting the data to the host system. When end-of-file is detected, the card-image file is closed. One READ command is required to read each card-image file.

The READ command is only valid if the SYCOM program is properly logged-on to the host system. It is possible to override this feature by entering the RY CR (ready card reader) command to the SYCOM program.

All non-graphic characters are replaced with the question mark (?) character.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.READ -----|  
      |-- <filename> |--| |-- <device type> |--| |-- % <comments> |--|
```

Semantics:

- filename** Specifies the label of the file. The SYCOM program uses the filename specified to locate the file on the designated device type. The filename must have no embedded blanks. The default file name is RJE/CARDS.
- device type** Specifies the type of device the SYCOM program is to read from. Allowable entries are DISK or blank (card reader device). The default device is CARD. If the pack-id is included in the filename, DISK is assumed as the device type.

Examples:

- .READ SOURCE % Card device is assumed.
- .READ USERPACK/SOURCE/FILE % Disk device is assumed since
% there are two slash (/) characters.
- .READ DISK/FILE DISK % Disk device is required since
% there is one slash (/) character.
- .READ % The filename defaults to RJECARDS
% and device type defaults to
% CARD.READER.

RESTART

The RESTART command causes the SYCOM program to reinitialize, break the data communication connection to the other system, flush the PUTQUEUE, and go into a standby state and wait for a connection to be initiated, by entry of either the ONLINE or CONNECT command to the SYCOM program.

If a schedule number is supplied, the SYCOM program re-establishes the connection with the other system and restarts the file transfers beginning with the supplied schedule number. Only the last 99 PUT commands in the PUTQUEUE can be restarted. Therefore, that the schedule numbers available for restarting are between the current PUT schedule number less 99, and the current PUT schedule number. If a minus sign (-) character appears in front of the schedule number, then the SYCOM program begins transferring all of the files in the PUTQUEUE starting with the current PUT schedule number less the number specified in the RESTART command.

If a plus sign (+) character appears in front of the schedule number, then the SYCOM program begins transferring all of the files in the PUTQUEUE starting with the current PUT schedule number plus the number specified in the RESTART command.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.RESTART ----->
|
|----- <integer> ----|
| |-- <minus sign (-) character> --|
| |-- <plus sign (+) character> ---|
|
|-- RESET -----|

>-----|
|-- % <comments> --|
```

Semantics:

minus sign The minus sign (-) character.
plus sign The plus sign (+) character.

integer Specifies the number (located in the PUTQUEUE and can be interrogated by entering the LIST SCHEDULE SYCOM command) to start the file transfers from.

RESET Causes the SYCOM program to disconnect and log-off all users.

Examples:

.RESTART % Flushes the PUTQUEUE and goes into Idle state.
.RESTART 12 % Re-establish connection and restart file transfer
% beginning with the schedule number 12 in the
% PUTQUEUE.
.RESTART -5 % Re-establish connection and restart file transfers
% beginning with the last five file transfers in the
% PUTQUEUE.
.RESTART +3 % Re-establish connection and restart file transfers
% beginning skipping three file transfers in the
% PUTQUEUE.
.RESTART RESET % Disconnects and logs-off all users.

RY (Ready Device)

The RY command informs the SYCOM program that the specified peripheral device is currently ready to send or receive data. The SYCOM program also notifies the server-system that the peripheral device is ready to send or receive data.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.RY ----- CR -----|  
      |--- CP ---|   |--- Z <comments> ---|  
      |--- LP ---|
```

Semantics:

- CR Ready the card reader.
- CP Ready the card punch.
- LP Ready the line printer.

Examples:

- .RY CR
- .RY CP Z Must be entered if punch output is from the
 Z B 1000 RJE/CONTROLLER program.
- .RY LP

SIGNAL

The SIGNAL command causes the SYCOM program to display or change the current SYCOM pseudo-signal character. This facilitates communication with more than one SYCOM program. If a new pseudo-signal character is not specified in the SIGNAL command, the current pseudo-signal character is displayed. If SMCS is executing and the SYCOM user is signed on to the SYCOM program, changing the SYCOM pseudo-signal character only applies to the one user, leaving the SYCOM pseudo-signal characters for the other users unchanged.

The default character for the SYCOM pseudo-signal character is the period (.) character or whatever character that is defined as the pseudo-signal character in the parameters file at beginning-of-job. The pseudo-signal character reverts back to the default pseudo-signal character after a station is signed off the SYCOM program or when the SYCOM program goes to end-of-job. A station can be signed off the program by entering the BYE command to the SYCOM program or the *OFF command to the SMCS. The pseudo-signal character is not required when a terminal is not connected to a server-system.

The SYCOM pseudo-signal character is required when entering SYCOM program commands to the SYCOM program on the system ODT and the system ODT is not connected to the remote system; otherwise, the message is sent to the remote system ODT.

Syntax:

```
.SIGNAL -----|  
      |--- <pseudo-signal character> ---|  |-- % <comments> --|
```

Semantics:

pseudo-signal character The new pseudo-signal character for the current SYCOM user.

Examples:

```
.SIGNAL  
.SIGNAL #
```

SIZE

The SIZE command causes the SYCOM program to display or change the maximum message buffer size for transmitting and receiving purposes. The SYCOM program then informs the server-system of this change with a special system control type message. This command applies only to the RJE function of SYCOM.

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the value of this parameter.

Syntax:

```
.SIZE -----|  
  |-- = --| |-- <integer> --| |-- % <comments> --|
```

Semantics:

integer A 3-digit number which assigns the RJE buffer size. The value must be between 078 to 400, inclusive.

Examples:

```
.SIZE=400        % Equal sign (=) character is optional.  
.SIZE 132  
.SIZE            % Causes the SYCOM program to display  
                 % the current RJE buffer size.
```

STOP

The STOP command causes the SYCOM program to immediately shut the system down. No further input messages are accepted and the SYCOM program goes to end-of-job. Terminals are not logged-off, virtual terminals are not disconnected, nor is an intersystem connection disconnected. Also, printer listings of the DISK_TRACE file and/or a SYCOM program dump are generated, if requested.

This command can only be entered from a remote ODT or the system ODT.

Syntax:

```
.STOP -----|  
  |-- D[IT] |--|  |-- % <comments> |--|  
  |-- xT  |--|
```

Semantics:

- D Causes the SYCOM program to initiate a dump.
- DT Causes the SYCOM program to initiate a dump and a list trace.
- xT Causes the SYCOM program to initiate a list trace. The "x" character represents any non-blank character except the "D" character.

Examples:

- ```
.STOP
.STOP D % Causes a dump.
.STOP DT % Causes a dump and list trace.
.STOP IT % Causes a list trace. The "I" character represents
% any non-blank character except the "D" character.
```

### SV (Save Device)

The SV command directs the user-system to inform the server-system to cease transmitting the card punch or line printer data stream and to save the entire file at the server-system. The SV command permanently inhibits the transmission of any further card punch or line printer data streams until notified by the user-system.

This command can only be entered from a remote ODT or the system ODT.

#### Syntax:

```
.SV ----- CP -----|
|---- LP ---| |-- Z <comments> --|
```

#### Semantics:

CP                    Save the card punch.

LP                    Save the line printer.

#### Examples:

```
.SV CP
.SV LP
```

## SYSTEMID

The SYSTEMID command displays or changes the current value of SYSTEMID. If a new SYSTEMID is entered, then it is sent to the remote system via the appropriate SYCOM control message. If the systemid has not been entered, the system hostname, entered by the MCP "FN" command will be used as the SYSTEMID.

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the value of this parameter.

### Syntax:

```
.SYSTEMID -----|
 |
 |----- <identifier> -----|
```

### Semantics:

<identifier> An identifier of up to seventeen characters containing only alphanumeric characters used to identify the new systemid.

### Examples:

```
.SYSTEMID
.SYSTEMID AARDVARK
```

## TRACE

The TRACE command provides the facilities for tracing all SYCOM input/output messages which are maintained in a file labeled DISK\_TRACE. The command is only valid if entered from the system ODT or a remote ODT station. When TRACE is entered without an attribute, the current status of the trace option is returned.

The disk tracing facility is used in conjunction with Program Switch 2. If Program Switch 2 is equal to 0, no data is traced, and only intersystem SYCOM program control messages are traced. If Program Switch 2 is equal to 15, all data is traced. If Program Switch 2 is equal to a number between 0 and 15, then n lines of data are entered into the trace file, where n is the assigned value of Program Switch 2.

The size of the DISK\_TRACE file can be controlled by the parameters TRACEAREAS, TRACEBA, and TRACERB, which default to 20, 10, and 4, respectively.

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the value of this parameter.

### Syntax:

```
.TRACE -----|
|--- CN ---| |-- Z <comments> --|
|--- CFF ---|
```

### Semantics:

ON Starts the tracing function.

OFF Stops the tracing function.

### Examples:

```
.TRACE Z Causes the SYCOM program to display the
Z current status of the TRACE boolean.
.TRACE ON
.TRACE OFF
```

Note that the disk tracing facility is used in conjunction with program switch 2. If program switch 2 is equal to zero, then only inter-system control messages are traced. If program switch 2 is set to a non-zero value, then a number of lines of data equal to the setting of program switch 2 is traced in addition to the inter-system control messages.

## USER

The USER command assigns the specified usercode/password as the default usercode/password for file transfers when none is specified in a COPY, PUT, or FETCH command. The usercode/password is not validated when entered but is forwarded to the other system when a CONNECT command is processed.

If no usercode/password is supplied, the station is disconnected (if connected) and the usercode is removed from the SYCOM program users table. This allows the user the ability to be logged-off the SYCOM program without signing off. The BYE SYCOM program command signs off the SYCOM user and detaches the station from the SYCOM program.

The usercode and password can be up to 17 characters each for communication with non-B1000 systems.

### Syntax:

```
.USER] ----->
 |-- = --| |-- <usercode> -----|
 |---- /<password> ----|

>-----|
 |-- % <comments> --|
```

### Semantics:

**usercode**        A symbol which consists of from 1 to 17 characters to identify the user. The B1000 computer system is limited to an 8 character usercode.

**password**        A unique string of characters used for identification purposes to meet security requirements.

### Examples:

```
.USER ODT
.US EDT/USER
.US % If a user is logged on to the SYCOM program,
 % then the user is logged off. Also, if the
 % station is connectec, then the station is
 % automatically disconnected. The station
 % remains attached to the SYCOM program.
```

## WAIT

The WAIT command is used to interrogate or change the status of the WAIT parameter.

This parameter can only be interrogated by a workstation. A remote ODT or the system ODT can change the value of this parameter.

### Syntax:

```
.WAIT -----|
 |-----| |-----|
 |----- ON -----| |----- Z <comments> -----|
 |-----| |-----|
 |----- OFF -----|
```

### Semantics:

ON                    Turns the wait parameter on.

OFF                   Turns the wait parameter off.

### Examples:

```
.WAIT
.WAIT ON
.WAIT OFF
```

## WHAT

The WHAT command causes the SYCOM program to display the current status of program processing activity. The reply message from the SYCOM program is variable in content and length to reflect as many conditions as possible. If a file is in the process of being transferred, its name and the number of records transferred are displayed.

### Syntax:

```
.WHAT -----|
--- U ---			
--- S ---			
--- = ---			
```

### Semantics:

- U Appends the output of a "LIST USERS" command to the normal output of the "WHAT" command.
- S Appends the output of a "LIST SCHEDULE" command to the normal output of the "WHAT" command.
- = Appends the output of both a "LIST USERS" and a "LIST SCHEDULE" command to the normal output of a "WHAT" command.

### Examples:

```
.WHAT
.WHAT U
.WHAT S
.WHAT =
```

### SMCS Command String from User-System

This command syntax allows a terminal user on the user-system to give commands to the server-system SMCS for processing. Commands which are entered from a terminal and prefixed with a signal character of asterisk (\*) are intercepted by the network controller and are immediately sent to the SMCS for processing. Consequently, the entire range of SMCS commands is available to a SYCOM user with this command syntax.

#### Syntax:

```
<signal character> ---- <SMCS command> -----
 |--- <attribute> ---|
```

#### Semantics:

signal character The signal character assigned to SMCS.

SMCS command Any SMCS command which is valid from a user-system terminal.

attribute Any attribute that is required by the SMCS command.

#### Examples:

```
*STATUS
*FILE DMPALL
```

### SMCS MCP Command String from User-System

This command syntax initiates directives to the user-system MCP for processing. By prefixing the command string with the signal character of the local SMCS, the network controller message switches this command string to the user-system SMCS which in turn zips the string to the MCP for processing under the following rules, assuming that the terminal is a remote ODT. If the ZIP option is included or the user is not logged-on, then the command string is zipped unsecured; otherwise, the command string is zipped secured.

#### Syntax:

```
<signal character> ----- <MCP command> -----|
| - Z[IP] -| | - <attributes> -|
```

#### Semantics:

signal character The signal character assigned to the SMCS.

Z[IP] Used by the SMCS to zip the MCP command as an unsecured control string.

MCP command Any MCP command which is valid from a work station. Refer to Appendix F in this manual, or a list of these MCP commands.

attribute Any attribute required by the MCP command.

#### Examples:

```
*Z PC DMPALL
*Z EX DMPALL
```

### SMCS Command String from Server-System

This command syntax allows a terminal user on the user-system to give commands to the server-system SMCS for processing. Each command entered must be prefixed with the current signal character for that SMCS and that signal character must differ from that of the user-system SMCS. All messages with this signal character are intercepted by the network controller and immediately sent to the SMCS for processing. Consequently, the entire range of SMCS commands is available to a SYCOM user. This command syntax is only valid when an SMCS is running in the server-system.

#### Syntax:

```
<signal character> ---- <SMCS command> -----|
|--- <attributes> ---|
```

#### Semantics:

signal character The character assigned to the server-system SMCS.

SMCS command Any SMCS command which is valid from a user-system terminal.

attribute Any attribute that is required by the SMCS command.

#### Examples:

```
% FILE CMPALL
% STATUS
```

### SMCS MCP Command String from Server-System

This command syntax allows certain MCP commands to be sent to a server-system for processing. These MCP commands can be entered from either the user-system ODT or from one of its terminals. All replies to these commands are returned to the user-system SYCOM program which causes them to be displayed on an appropriate device. Certain MCP commands require a usercode/password, prefixed to the desired MCP command, which is known to the server-system. This command syntax is only valid when an SMCS is running in the server-system.

#### Syntax:

```
<signal character> ----->
 | | | |
 |- Z[IP] -| |- US <usercode> -----|
 |-- <password> ---|

>----- <MCP command> -----|
 | |
 |--- <attributes> ---|
```

#### Semantics:

- signal character The signal character assigned to the server-system SMCS.
- Z[IP] Used by the SMCS to zip the MCP command as an unsecured control string.
- usercode A symbol which consists of from one to eight characters that identifies the user.
- password A unique character string used for identification purposes to meet security requirements.
- MCP command Any MCP command that is valid from an ODT or terminal.
- attribute Any attribute required by the MCP command.

#### Examples:

```
$ MX
$ WY
$ US NY/USER PD =/=
```

Table 4-1 contains a summary of the functions of the SYCOM commands. Refer to the command descriptions in this section, for command syntax and option details.

Table 4-1. Summary of SYCOM Commands

| Command<br>----- | Station<br>----- | Function<br>-----                                            |
|------------------|------------------|--------------------------------------------------------------|
| ABORT            | Any              | Abort a PUT or FETCH file transfer.                          |
| ADD              | Any              | File transfers (PUT or FETCH).                               |
| BUFFER           | Any*             | Interrogate or change the buffer size for file transfers.    |
| BYE              | Any              | Station log-off from SYCOM.                                  |
| CL               | ODT              | Clear peripheral device.                                     |
| CONNECT          | Any              | Request establishment of a terminal to a server-system SMCS. |
| COPY             | Any              | File transfers (PUT or FETCH).                               |
| CP               | ODT              | Inform SYCOM of Not Ready peripheral.                        |
| DEBUG            | Any*             | Start/stop input/output message tracing.                     |
| DISCONNECT       | Any              | Request for terminal disconnect from the server-system SMCS. |
| EOJ              | ODT              | Causes SYCOM to go into an orderly shutdown.                 |
| FETCH            | Any              | File transfers (receive a file from other system).           |
| GROUPSIZE        | Any*             | Sets groupsize for file transfers.                           |

|          |      |                                                         |
|----------|------|---------------------------------------------------------|
| LIST     | Any  | Display selective system information.                   |
| LOG      | Any* | Display a list of Data Comm error counts.               |
| LP       | ODT  | Inform SYCOM that a line printer is Not Ready.          |
| OFFLINE  | ODT  | Cause SYCOM to break the connection to a server-system. |
| ONLINE   | ODT  | Establish a server-system connection.                   |
| PHONE    | Any* | Establish or clear phone number.                        |
| PROGRAM  | Any* | Set or interrogate value of program parameter.          |
| PUT      | Any  | File transfers (place a file on the other system).      |
| QS       | Any* | Change network controller queue depth.                  |
| READ     | ODT  | Open and transmit card input file.                      |
| RESTART  | ODT  | Disconnect line, initialize, and standby.               |
| RY       | ODT  | Inform SYCOM of available peripheral.                   |
| SIGNAL   | Any  | Change the SYCOM signal character.                      |
| SIZE     | Any* | Change maximum message block size.                      |
| STOP     | ODT  | Cause SYCOM to shut down immediately.                   |
| SV       | ODT  | Stop transmitting and save file.                        |
| SYSTEMID | Any* | Interrogate or change the value of SYSTEMID.            |

|       |      |                                                        |
|-------|------|--------------------------------------------------------|
| TRACE | Any* | Start/stop SYCOM I/O disk tracing.                     |
| USER  | Any  | Log a user on to the SYCOM program.                    |
| WAIT  | Any* | Interrogate or change the value of the wait parameter. |
| WHAT  | Any  | SYCOM status inquiry.                                  |

-----

|      |   |                                                                                                                 |
|------|---|-----------------------------------------------------------------------------------------------------------------|
| Any  | = | Any SYCOM station                                                                                               |
| Any* | = | Any SYCOM station can interrogate the value of this command. Only the system ODT or a remote ODT can change it. |
| ODT  | = | The system ODT or a remote ODT.                                                                                 |

## OPERATING PROCEDURES

### GENERAL

This section provides system and terminal operating procedures used for the operation of a SYCOM system with and without local terminals and/or in conjunction with the Supervisory Message Control System (SMCS). System operating procedures are slanted toward operation of SYCOM for the purpose of Remote Job Entry (RJE) type functions and the associated system control functions. These procedures present all aspects of SYCOM operations from initialization thru controlled shutdown.

Local terminal operational procedures are separated into two (2) categories; terminals under SYCOM control and terminals under the control of the SMCS. These procedures provide coverage of all terminal operations within the scope of SYCOM) from sign-on to sign-off. Actual input/output message text is used in the examples to provide a feel for the ease of operation and what can be expected by a user of the system.

### SYSTEM OPERATING PROCEDURES

The system operational procedures presented here are intended to provide a User System operator with sufficient information for operation of an interactive system using SYCOM. Operation of the associated Network Controller is briefly discussed. In addition, suggestions on the use of certain basic SYCOM commands and their options are exemplified.

### PRELIMINARY PROCEDURES

Before the SYCOM system is executed, the user needs to be aware of a few system variables that must be considered. First there are "program switches" which may be used to invoke certain optional features. Second there is the "parameters" file which is used to maintain SYCOM's variable control functions and processing attributes. The following procedures must be performed to make the SYCOM system operational:

1. Set the SYCOM program switches as appropriate.
2. Prepare a parameters disk or card file as required. (optional)
3. Create a (SYSTEM)/MAKEUSER file, if required for file security.

4. Setup the MCP program name table. (Optional - for automatic MCP execution of Network Controller when SYCOM is executed)
5. Assign system program priorities.

#### Program Switches

Program Switch 0 Setting Program Switch 0 to 4 causes the SYCOM program to omit the display of beginning-of-job messages until the link to the remote system has been established. If the low-order two bits of the switch are set, the DEBUG/DISK\_TRACE trace facility will be activated from BOJ until the parameters file is read by the SYCOM program. As soon as the parameters file has been read, the values contained in the parameters file will be used to control the DEBUG/DISK\_TRACE trace facilities. This feature is intended to be used for debugging purposes.

Program Switch 1 If Program Switch 1 is set to 1, then the open-on-behalf-of file security mechanism will not be used for disk files. This will cause SYCOM to use the Mark 7.0 file security mechanisms. This switch is only examined when processing operator input. Changing program switch 1 to any value than 1 will restore the normal mode of operation with file security.

Program Switch 2 Program Switch 2 is only valid when the TRACE parameter is specified in the parameters file or the TRACE ON command is entered to the SYCOM program at the system ODT.

Setting Program Switch 2 to 0 causes the SYCOM program to trace intersystem SYCOM control messages only. Setting Program Switch 2 to 15 causes the SYCOM program to write all messages and data lines to the DISK\_TRACE file. Setting Program Switch 2 to a value between 0 and 15 causes the SYCOM program to write n number of data lines to the DISK\_TRACE file, where n is the values specified for Program Switch 2.

Program Switch 3 If Program Switch 2 is set to 3, then the output from the SERVER stations is included in the DISK\_TRACE file if the TRACE option is set or in the DEBUG file if the DEBUG option is set.

Program Switch 4 Setting Program Switch 4 causes the SYCOM program to suppress the file transfer messages displayed on the system ODT. The value of Program Switch 4 determines which message is to be suppressed. The messages that can be suppressed follow.

| Message Number | File Transfer Message                  |
|----------------|----------------------------------------|
| 1              | FILE "<filename>" START SENDING        |
| 2              | FILE "<filename>" START RECEPTION      |
| 3              | FILE "<filename>" SENT <n> RECORDS     |
| 4              | FILE "<filename>" RECEIVED <n> RECORDS |

Table 3-1 shows the possible combinations for suppressing the file transfer messages.

**Table 3-1. Combinations of File Transfer Messages**

| Value of Program Switch 4 (in hex) | Message Suppressed (by message number) |   |   |   |
|------------------------------------|----------------------------------------|---|---|---|
|                                    | 1                                      | 2 | 3 | 4 |
| a0a                                |                                        |   |   |   |
| a1a                                |                                        |   |   | X |
| a2a                                |                                        |   | X |   |
| a3a                                |                                        |   | X | X |
| a4a                                |                                        | X |   |   |
| a5a                                |                                        | X |   | X |
| a6a                                |                                        | X | X |   |
| a7a                                |                                        | X | X | X |
| a8a                                | X                                      |   |   |   |
| a9a                                | X                                      |   |   | X |
| aa a                               | X                                      |   | X |   |
| aba                                | X                                      |   | X | X |
| aca                                | X                                      | X |   |   |
| ada                                | X                                      | X |   | X |
| aea                                | X                                      | X | X |   |
| afa                                | X                                      | X | X | X |

The "X" indicates that the message is suppressed.

If Program Switch 4 equals 202, then no messages are suppressed. If Program Switch 4 equals 2F2, then all the messages are suppressed.

Program Switch 4 only applies to the local SYCOM program and does not affect the file transfer messages displayed by the SYCOM program on the other system.

Program Switch 5 Setting Program Switch 5 to 5 causes the SYCOM program to automatically initiate a memory dump and print the DISK\_TRACE file whenever the SYCOM system appears to out of phase.

Setting Program Switch 5 to 5 also causes the SYCOM program to automatically print the DISK\_TRACE file when the file becomes full. The SYCOM program continues writing trace records beginning at record number 1 in the DISK\_TRACE file.

Program Switch 6 Setting Program Switch 6 to 6 causes the SYCOM program to display the statistics of the SYCOM log on the system ODT whenever the DISK\_TRACE file becomes full. The SYCOM program begins writing trace records to the DISK\_TRACE file at record number 1.

Setting Program Switch 6 to 7 causes the SYCOM program to automatically write the SYCOM log to the program INQ file when the DISK\_TRACE file becomes full.

Program Switch 7 Setting Program Switch 7 to 1 causes an entry in the statistics file to be generated for files that have been received from a remote system.

Setting Program Switch 7 to 2 causes an entry in the statistics file to be generated for files that have been transmitted to a remote system.

Setting Program Switch 7 to 3 causes an entry in the statistics file to be generated for any file transfer that has occurred (i.e. both sending and receiving).

#### Workfile Naming Conventions

The family name and pack id of the SYCOM workfiles (NEWPARAMS, PUTQFILE, INQ, OUTQ, STATISTICS, RJE\_Q, LINK\_Q, DISK\_TRACE, and OLDTRACE) are determined from the family name and pack id of the PARAMETERS

The family name and pack id of the parameters file are determined as follows:

1. If any file equation has been made for the PARAMETERS file by the operator, then the actual name and/or pack id specified will be used. For example:

- a. EX SYCOM FILE PARAMETERS PID USER.DISK;
- b. EX SYCOM FILE PARAMETERS NAME USER.PARMS;
- c. EX SYCOM FILE PARAMETERS NAME USER.PACK/HUB/PARMS;

The actual files names for the PARAMETERS file resulting from the file equations shown above are as follows:

- a. USER.DISK/<sycom>/PARAMETERS
- b. USER.PARMS
- c. USER.PACK/HUB/PARMS

2. If no file equation has been made for the PARAMETERS file by the operator, then the pack id and the family name will be taken from the actual name of the SYCOM program. For example:

- a. EX SYCOM
- b. EX HUB.SYCOM
- c. EX DC/SYCOM10/SA1

The actual names for the PARAMETERS file resulting from the execution statements above are as follows:

- a. SYCOM/PARAMETERS
- b. HUB.SYCOM/PARAMETERS
- c. DC/SYCOM10/PARAMETERS

All workfiles will be changed by SYCOM to have the same family name and pack id as the parameters file. Thus, many copies of SYCOM may be run on a single machine and the workfiles may be located on different packs as desired.

Since the only way to force printer backup files to a user disk is by using the MCP "BD" command, SYCOM cannot force the DEBUG or LIST files to a user pack. The family name of these files is changed to the same family name as the other workfiles so that there will be no confusion over the labelling of printer files from multiple copies of SYCOM.

#### Creating the SYCOM Parameters File

When the SYCOM program is executed for the first time or if the SYCOM PARAMETERS file is not on disk, a parameters file must be created. At beginning-of-job the SYCOM program first checks for the parameters file on disk. The actual label of the parameters file is determined as described above. If the PARAMETERS file is present and of the correct version, SYCOM uses the parameters

specified in the disk file. If the disk file is not present, the SYCOM program checks for a card file labeled NEWPARAMS. If the card file is not present the program looks for a disk file labeled NEWPARAMS. If the card file or disk file labeled NEWPARAMS is present, then the SYCOM program uses the parameters specified in the card file to create the PARAMETERS disk file. If the card file and disk file labeled NEWPARAMS are not present, the SYCOM program prompts the user to enter the parameters or "END" (since all parameters have a default value) from the ODT.

Once the parameters have been entered, the SYCOM program creates a disk file labeled PARAMETERS for use the next time the program is executed. The PARAMETERS file does not need to be recreated unless the file is removed. removed or requires permanent changes. The parameters may be changed or interrogated for the current execution of SYCOM by entering the parameter name and a value in an ACCEPT message. To create a new PARAMETERS file, the following steps should be followed:

1. Make the required changes to the NEWPARAMS file.
2. Remove the PARAMETERS disk file.
3. Terminate the SYCOM program.
4. Re-execute the SYCOM program.

Refer to SYCOM Parameters, in Section 6 page 15, for a description of the SYCOM parameters.

### Preparing the NEWPARAMS Card File

The NEWPARAMS file can be a card file. The following is an example of a NEWPARAMS card file.

?DATA NEWPARAMS

```
SYSTEMID = USER1 %
BUFFER = 1100 % Allows 3 code records per transmission.
DEFAULTPSEUDOSIGNAL $ % Use "$" instead of "."
GROUPSIZE = 1 % Default is 9
NOFS % For Host RJE
NOFTP %
NOPRCGRAM %
QUEUESIZE = 6 % The default value is 4.
WAIT % The default is no wait.
TIMECUTS = 0 % Default is 11. Eliminate message
DEBUG % Default is false
TRACE %
TRACEAREAS = 25 % The default value is 20.
TRACEBA = 8 % The default value is 10.
TRACERB = 5 % The default value is 4.
END % Optional.
```

?END

-----1-----2-----3-----4-----5-----6-----

The question mark (?) character in the above example represents an invalid character for 80-column cards. The percent sign (%) character causes the SYCOM program to stop scanning the record. This allows user comments to be included in a parameter record.

The above example does not show all of the available SYCOM parameters. If a parameter is not specified in the card file, then the SYCOM program uses the default value for the parameter.

### Preparing the NEWPARAMS Disk File

The NEWPARAMS disk file can be created by using the CANDE program. The parameters can be entered into the NEWPARAMS file in free-form format. The percent sign (%) character can be used to denote the beginning of comments for documentation purposes. The following CANDE-created disk file shows what the NEWPARAMS file can look like. If CANDE is used to create the NEWPARAMS file, it may be of type DATA or SEQDATA.

The following is an example of a NEWPARAMS disk file.

```
%
% Sample NEWPARAMS files
%
% Lower-case entries are allowed in parameters file for
% parameters.
%
systemid = B1885 % Special characters are not allowed.
 % If this parameter is not specified,
 % the SYCOM program will use the MCP
 % HOSTNAME.
 %
defaultpseduosignal & % Controls the SYCOM pseudo-signal
 % character. Default is ".".
 %
groupsize = 5 % Controls the number of files that
 % may be sent at one time when sending
 % an entire family of files.
 % Default is 9.
 %
buffer = 1950 % The maximum buffersize is 1950.
 %
inbuffers = 5 % Number of buffers assigned to
 % file "IN" at BOJ. Default is 3.
 %
inbuffers = 5 % Number of buffers assigned to
 % file "IN" at BOJ. Default is 3.
 %
program % Allows program-to-SYCOM functions.
 % Default is true.
 %
% noprogram % Inhibits program-to-SYCOM.
 % Default is false.
 %
% noftp % Inhibits file transfer.
 % Default is false.
 %
queuesize = 8 % Controls the number of messages
 % that can be written in the "LINK"
 % file. Range is from 1 to 15.
 % Default is 4.
 %
timeouts = 11 % If n timeouts occur, then the
 % remote SYCOM is consider off-line.
 % Default is 11.
 %
% wait % Prevents SYCOM from going to EOJ
 % when a disconnect message is
 % received. Default is false.
 %
 nowait % Allows SYCOM to go to EOJ when
 % when disconnect message is
 % received. Default is true.
 %
% trace % Allows disk tracing.
```

```
 Z Default is true.
 Z
notrace Z Inhibits disk tracing.
 Z Default is false.
 Z
tracerb = 4 Z Records-per-block for the trace
 Z file. Default is 4.
 Z
traceba = 10 Z Blocks-per-areas for the trace
 Z file. Default is 10.
 Z
traceareas = 20 Z Number of areas for the trace
 Z file. Default is 20.
 Z
% debug Z Allows printer tracing.
 Z Default is false.
 Z
% nofs Z File separators are allowed by
 Z default. This parameter is used
 Z in conjunction with the Host RJE
 Z program which does not allow file
 Z separators.
 Z
 Z Optional. Signifies end of
 Z parameters.
 Z
-----1-----2-----3-----4-----5-----6-----*
```

### Entering the Parameters Through the ODT

If the SYCOM program does not find a disk file labeled PARAMETERS, a card file labeled NEWPARAMS, and a disk file labeled NEWPARAMS, the following message is displayed on the ODT.

```
% SYCOM =<job #> ENTER OPTIONAL SYCOM PARAMETERS OR END
% SYCOM =<job #> ACCEPT
```

The SYCOM program is then ready to accept parameters from the ODT. The parameters are entered in free-form format and are terminated by an "END" parameter.

The parameters must be entered with ACCEPT entries in the following format:

```
<SYCOM job #> AX <parameter keyword> [=<value>]
 [<parameter keyword> [=<value>]....
 END
```

Multiple ACCEPT entries are allowed. The sequence is terminated by the "END" parameter.

The SYCOM program checks each parameter for syntax and validity. Any incorrect entry invalidates all entries in that ACCEPT entry. Correct the error and re-enter the ACCEPT entry. If the default value for a parameter is desired, it is not necessary to include it in the ACCEPT entries.

The parameters are described in Section 4.

#### Creation of the (SYSTEM)/USERCODE File

The system utility program "MAKEUSER" is used to create and maintain the (SYSTEM)/USERCODE file. This file contains a list of all valid usercode/password combinations for the system. Refer to the "Systems System Software Operational Guide" for detailed descriptions of the file and its creation.

#### Setup MCP Program Name Table

In order for SYCOM's Network Controller to be automatically executed by the system, it is necessary to establish the identity of this Controller in the "MCP Program Name Table". To do this is simple, enter the following MCP command:

```
--> CM C USER/NC<etx>
 NETWORK CONTRLLER CHANGED FROM "" TO "USER/NC"
```

The controller name "SYCOM/NC" is of course a user optional name and is therefore only representative. The actual Controller name selected by the user would be substituted as required.

To verify that the correct Network Controller I/D is in the MCP Name Table ("C" slot), the following MCP command can be entered:

```
--> WW C<etx>
 C = "USER/NC"
```

#### Assigning Program Priorities

For optimum system response, it is necessary that the system's Network Controller and SYCOM run at a "high" priority. Therefore, to ensure that the MCP automatically executes these programs at their optimum priorities, the following MCP commands are used to permanently assign a priority of "15" to the Network Controller and a priority of "12" to SYCOM:

```
--> MO USER/NC PR 15<etx>
 "USER/NC" MODIFIED
```

```
--> MO SYCOM PR 12<etx>
 "SYCOM" MODIFIED
```

### Execution of SYCOM; Without SMCS

To use the SYCOM system without the benefit of a SMCS, the system operator simply enters the execute SYCOM command as shown below. The MCP automatically executes the associated Network Controller program (providing one is in the "C" slot of the MCP Name Table) when the first remote file open is initiated for the controller. Both SYCOM and the Network Controller should be run at a priority which ensure adequate response times to the user of the system (see suggested priority values below).

It is the option of the user whether or not SYCOM is executed with a "usercode". If it is desired to force SYCOM's work files off the system disk, then it may be executed with a privileged usercode whose default pack-name is the desired disk-cartridge or disk-pack. Otherwise, SYCOM must be executed in the manner shown in the example below:

```
--> EX SYCOM<etx>
```

```
SYCOM =<job #> BOJ. PP=12, MP=12 TIME = 08:14:51.2
```

During the beginning of the initialization phase if SYCOM cannot locate the "SYCOM/PARAMETERS" file or disk, it attempts to obtain new parameters from a disk file labeled "SYCOM/NEWPARAMS" or from the card reader ("SYCOM/NEWPARAMS" file). Should the disk file not be available or the card reader "not ready", or if there is no card reader, then SYCOM will lapse into the mode of accepting the new parameters from the system ODT. When SYCOM is in this mood, the following messages are displayed in the ODT:

```
% SYCOM =<job #> ENTER SYCOM OPTIONAL PARAMETERS or "END"
SYCOM =<job #> ACCEPT.
```

The system operator can then enter the appropriate parameters to alter the system's "default parameters" (refer to Appendix G, "Parameters File Attributes"). The entry procedure is presented with "Entering The Parameters" in this section. Although the "SYSTEMID" parameter is not required, in some instances, it is strongly suggested that a representative "system identifier" be entered with the parameters at this time.

Once this file is established on disk, it need not be recreated unless it is lost or requires changes. Changes to the file can be made using either of the two methods identified. However, again, it is necessary to "remove" the existing parameters file from disk. Additionally, this requires SYCOM to be shutdown and re-executed if it is currently running; else all is in order.

When SYCOM opens its Remote Files, the MCP knows whether or not a Network Controller is running and takes the appropriate action. That is, if a Controller is running, then the remote file opens take place and the system becomes operational. If a Controller is "not" currently running, then if the Controller's name is in

the "C" slot of the MCP Program Name Table, the MCP automatically executes the named Controller program and the following messages are displayed at the ODT:

```
EX SYCOM/NC PT;
SYCOM/NC =<job #> BOJ. PP=15, MP=15 TIME = 08:15:08.7
```

If there is no Controller name entered in the MCP Program Name Table, then the following message is displayed:

```
REMOTE FILE "TERMINALS" NEEDS A CONTROLLER
```

The system operator must then execute the appropriate Network Controller by entering the following command:

```
--> EX SYCOM/NC<etx>
```

```
SYCOM/NC =<job #> BOJ. PP=15, MP=15 TIME = 08:16:01.2
```

The MCP displays the above "BOJ" message, permits the Remote File opens to occur and the system is operational. When this condition exists, the following optional message is displayed on the system ODT:

```
% SYCOM =<job #> RUN DAY = THURSDAY 17 AUG 79 AT 08:23:31
```

Upon completion of SYCOM's initialization phase and a successful connection established to a default Server System, the following messages will be displayed on the User ODT:

```
% SYCOM =<job #> **** SYCOM IX.0.2 (30 JUL 79 14:53) ***
NOW ONLINE TO REMOTE HOST = SERVER1
```

### Execution of SYCOM; With SMCS

When SYCOM runs in conjunction with the SMCS, it is automatically executed by SMCS thru its "job spawning" facility. This occurs when the "first" terminal user "SIGNS-ON" or "PASS's" to SYCOM from a local terminal. Or, the system operator may start SYCOM from the system ODT by entering the SMCS command "START SYCOM". SYCOM executed by any other method will only cause operational problems (i.e., 2 copies of SYCOM executing at the same time).

Prior to SYCOM's operation under the SMCS, an entry must be made in the "SMCS/JCBS" file for SYCOM. A sample entry is included in the "SMCS Software Operational Guide" for this purpose under the heading "Editing The JOBS File".

If during the initialization phase, SYCOM cannot locate the file "SYCOM/PARAMETERS", it attempts to obtain new parameters from the disk file "SYCOM/NEWPARAMS" or the card reader (SYCOM/NEWPARAMS file). Should this either file not be present, or if there is no card reader, then SYCOM commences to accept new parameters input from the ODT. To signal the system operator of this requirement, the following messages are displayed:

```
% SYCOM =<job #> ENTER SYCOM OPTIONAL PARAMETERS or "END"
SYCOM =<job #> ACCEPT.
```

The system operator must then enter the appropriate parameters to override the system's "default parameters" (refer to Appendix G, "Parameters File Attributes"). The parameter entry procedure is presented with "Entering The Parameters"; this section. Although the parameter "SYSTEMID" is not required, for some instances, it is strongly suggested that a representative "system identifier" be entered with the parameters at this time.

Also during the initialization phase, the following optional message is displayed on the User ODT:

```
% SYCOM =<job #> RUN DAY = THURSDAY 17 AUG 79 AT 08:23:31
```

Upon completion of initialization and the successful connection to a default Server System, the following messages are displayed on the User ODT:

```
% SYCOM =<job #> **** SYCOM IX.0.2 (30 JUL 79 14:53) ***
NOW ONLINE TO REMOTE HOST = SERVER1
```

### Common System Operations

Once SYCOM has been executed, from either the ODT or spawned by the SMCS, most operating procedures are common to both modes of operation. These common operating procedures are oriented to ODT

input in the form of SYCOM parameters, SYCOM commands, and MCP directives. Following then are the common operating procedures, presented in the sequence they are most likely to occur, under individual headings.

### SMCS Signal Characters

In order for a SYCOM systems network to function effectively, it is a requirement that the SMCS "signal character" for each system in the network to be a different character. This can be easily accomplished by mutual agreement when the SYCOM system network is designed. Each system's SMCS should have a specific character which is made known to all system users. Obviously, if the SMCS is not utilized, this is not necessary.

If during actual system operations it is found that a SMCS signal character is undesirable for any reason, it can be changed while online. This can be accomplished by entering a "signal character change command" from either a local or remote source. For actual details, refer to the "SMCS Software Operational Guide".

### Entering the Parameters

The parameter definitions listed below can be entered to SYCOM from the system ODT. To do so, they can be entered on several lines using the following "Accept" process:

```
--> <sycom job #> AX <parameter keyword> --- [=<value>] ----->

-----> [<parameter keyword> --- [=<value>]....] END
```

Each parameter is validated and the system operator is informed of any illegal parameters encountered during the parsing of the parameter data. If the parameters are specified in a disk or card file "SYCOM/NEWPARAMS", then they may also extend over several records (see example). Comments on any record must be preceded by a "Z".

**NOTE: All parameters have default values.**

### **BUFFER Parameter**

This parameter defines the size, in bytes, of the used in SYCOM communications with the other system. The buffer size should be large enough to handle the largest message, and possibly achieve a blocking factor for multiples of a record.

Syntax: BUFFER = <integer>

The default buffer size is 1950 (a compile time constant) and the maximum allowable size is 1950 bytes. (This parameter directly relates to file transfer buffer size and not RJE buffer size.) Be advised however, that this buffer size must be equal to or greater than the maximum record size to be transferred multiplied by 2, plus 2 needed for non-data record translations.

### **DEBUG Parameter**

This parameter indicates that the printer tracing mechanism is to be utilized to produce a record of SYCOM's I/O message activity. (The default is "no printer trace".)

Syntax: DEBUG

### **DEFAULTPSEUDOSIGNAL**

The DEFAULTPSEUDOSIGNAL parameter is used to permanently assign a value to the SYCOM pseudo-signal character. The first non-blank character encountered after the parameter is taken as the pseudo-signal character. (The default is ".").

Syntax: DEFAULTPSEUDOSIGNAL ----- <character> -----|

### **END Parameter**

Entering an END in the parameter string terminates the parameter input process.

Syntax: <sycom job #> AX <parameter string> END

## GROUPSIZE

The GROUPSIZE parameter is used to control the number of files that will be sent at one time when sending entire families of files. The SYCOM program will transmit a number of files equal to the groupsize and then check to see if another user has files to be sent and, if so, will transmit a number of files for the second user equal to the groupsize, before continuing with the second and subsequent groups. (The default is 9.)

## INBUFFERS

The INBUFFERS parameter controls the number of buffers that are assigned to the file "IN" at BOJ. The file "IN" is used for input when doing file transfer and increasing the number of buffers can speed up the file transfer process. (The default is 3.)

Syntax: INBUFFERS = <integer>

## NOFS Parameter

SYCOM will not send a "FS1" or "FS2" (file separator) to the other system when sending card images (relevant to RJE). (The default is false.)

Syntax: NOFS

## NOFTP Parameter

This parameter inhibits File Transfers if either system has this parameter specified. The purpose is clear, when specified, "no" File Transfers are to occur for this system. There is no quick way to overcome this user restriction. The only recourse is to shut-down SYCOM, remove the Parameters file and correctly enter the desired parameters. (The default is false.)

Syntax: NOFTP

## NOPROGRAM

The NOPROGRAM parameter inhibits program-to-SYCOM functions. (The default is false.)

Syntax: NOPROGRAM

### NOTRACE

The NOTRACE parameter inhibits tracing at beginning-of-job. (The default is true.)

Syntax: NOTRACE

### NOWAIT

The NOWAIT parameter allows SYCOM to go to end-of-job when it receives a disconnect message. (The default is true.)

Syntax: NOWAIT

### OUTBUFFERS

The OUTBUFFERS parameter controls the number of buffers that are assigned to the file "OUT" at BOJ. The file "OUT" is used for output when doing file transfer and increasing the number of buffers can speed up the file transfer process. (The default is 2.)

Syntax: OUTBUFFERS = <integer>

### PROGRAM Parameter

This parameter is specified when SYCOM is to perform program-to-SYCOM functions. The "record size" parameter limits the maximum length of these messages. (The default is true.)

Syntax: PROGRAM

### QUEUESIZE Parameter

This parameter is used to establish the initial QUEUESIZE default value. The value is changeable at any time during SYCOM operation via the "QS" command. The minimum value which can be specified is "4" and the maximum is "15". (The default is 8.)

Syntax: QUEUESIZE = <value>

### SYSTEMID Parameter

This parameter is used to specify the name of the User System. A system-id can be comprised of from one to seventeen characters.

Syntax: SYSTEMID = <system-id> (e.g., USER1)

If the SYSTEMID is not specified the MCP HOSTNAME will be used.

### TIMEOUTS Parameter

This parameter establishes an "interval" for the assumption that the remote system has gone "offline". A value of "0" inhibits this function. (The default is 11.)

Syntax: TIMEOUTS = <number>

### TRACE Parameter

The usefulness of this parameter is to specify "disk tracing" at SYCOM initialization. The default is no tracing initiated.

Syntax: TRACE

### TRACEAREAS

This parameter is also specified in conjunction with "TRACE". It is used to specify the "number of areas" for the disk trace file. The default value is "20".

Syntax: TRACEAREAS = <value>

### TRACEBA Parameter

This parameter is also specified in conjunction with "TRACE". It is used to specify the "blocks per area" for the disk trace file. The default value is "10".

Syntax: TRACEBA = <value>

### TRACERB Parameter

This parameter is specified in conjunction with "TRACE". It is used to specify the "records per block" for the disk trace file. The default value is "4".

Syntax: TRACERB = <value>

### WAIT Parameter

This parameter is used to prevent SYCOM from going to "end-of-job" when it receives a "disconnect message". SYCOM will disconnect and lapse into a "offline/idle" state. (The default is false.)

Syntax: WAIT



```
% <sycom>/PARAMETERS on disk.

queuesize = 4 % this is the default value. Maximum is 15
% QUEUESIZE (MAX_MESSAGES in SYCOM)
% controls the number of messages written
% to the LINK file. Its dependent variable
% MARKER_OFFSET, for determining when the
% "GOOD_RESULTS" mechanism is invoked, is
% (MAX_MESSAGES / 2) + 1
% The queuesize may be queried or changed
% run time by using the ".QS" command.
% Any new value will not be saved in the
% <sycom>/PARAMETERS file on disk.

timeouts = 11 % this is the default value. If this number
% of consecutive timeouts occur then the
% remote system is considered OFFLINE.

% trace % this is the default for DISK tracing.
% notrace % default = FALSE
% tracerb = 4 % default = 4 records per block
% % 4 * 132 = 528
% % 540 = 180 * 3
% tracerb = 8 % 8 records per block = 1056
% % 6 sectors at 180 bytes = 1080 bytes
% traceba = 10 % default = 10 blocks per area
% traceareas = 20 % default = 20 areas
% % 4 * 10 * 20 = 800 records in DISK.TRA

% wait % default = FALSE
% nowait % default = TRUE
% debug % default = FALSE
% noftp % default = FALSE, i. e. file transfer is allowed
% nofs % default = FALSE, i. e. file separators are sent.
% This parameters is used in conjunction with
% HOST/RJE (B1700/B1800) which currently will not
% accept file separators.

defaultpseudosignal. % this is the default (".")
% may be set to any non blank character

groupsize = 9 % this is the default (= 99/10) and is
% number of files sent in group with in a
% family of files being transferred.
% N.B. 99 is the size of the PUT QUEUE file
% If this parameter is set to 1 only one
% family member will be sent with checking
% for any other file transfer requests.
% If, however, it is set to the maximum
% value of 99 then no other file transfer
% requests will be honoured until the
% whole family has been transferred.
```

```
inbuffers = 3 % This is the default value. It can be set
 % to any other value and is the number of
 % buffers allocated to the file <sycom>/IN

outbuffers = 2 % This is the default value. It can be set
 % to any other value for the number of
 % buffers allocated to the file <sycom>/OUT

end % This is a required parameter if the
 % parameters are being entered from the ODT
 % It is the last parameter and is entered
 % to terminate the list of parameters.
 % It optional if the parameters are being
 % read from the <sycom>/NEWPARAMS file.
```

```
% N.B. The following parameters can have their value changed at run
% time by the SYCOM command following the parameter:-
```

```
% buffer buffer
% debug debug
% groupsize groupsize
% program program
% noprogram program
% queuesize qs
% systemid systemid
% trace trace
% notrace trace
% wait wait
% nowait wait
```

```
% Any SYCOM station can interrogate the value of any of the above
% parameters at run time with any of the above SYCOM commands. Only
% stations declared as ODT (or CHANGE'd by the SMCS) can alter any
% of the values. The altered value is valid until the next alteration
% or SYCOM ECJ. At next SYCOM BOJ the values will revert to their
% previously declared values contained in the <sycom>/PARAMETERS file
% To permanently change the values in the <sycom>/PARAMETERS file it
% must be removed and re-created.
```

```
% e. g. :- interrogation reply from SYCOM
% *p sycom .buffer RUN TIME TRANSMISSION BUFFER SIZE IS 1950, RJE SIZE = 400
% *p sycom .debug
% DEBUG OFF
% *p sycom .groupsize GROUP SIZE IS 9
% *p sycom .program PROGRAM INPUT ENABLED
% p sycom .qs QUEUE SIZE/OFFSET = 4/3
% *p sycom .systemid SYSTEM ID = "SLAVE3"
% *p sycom .trace TRACE ON 4 * 10 * 20 = 800 RECORDS
% *p sycom .wait WAIT OFF
%
```

### SYCOM Commands from the ODT

The system operator is permitted to communicate with SYCOM using the MCP "AX" facility. By entering certain commands (examples below), the operator is also permitted to communicate with application programs in a Server System; thru the "virtual terminal" facility of SYCOM. A typical command session might go as follows; ODT input is flagged with "-->" and indented lines are output reply messages:

--> <sycom job #> AX.WHAT<etx>

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; 2 FILES QUEUED;  
SENDING: (JH)/#665 (7901/15325);  
THIS STATION NOT CONNECTED  
LOCAL HOST = USER1, REMOTE HOST = SERVER1

--> <sycom job #> AX.US MY/USER<etx>

B1700/B1800 SYCOM IX.0.2 (30 JUL 79 14:53)  
USER MY LOGGED ON TO SYCOM AT 11:31:22.3 ON 22 AUG 79

--> <sycom job #> AX.FETCH \*DMPALL<etx>

FILE DMPALL START TRANSMISSION  
FILE "DMPALL" START RECEPTION  
FILE "DMPALL" RECEIVED  
FILE DMPALL END TRANSMISSION

--> <sycom job #> AX.PUT (MY)/#3152 (NO LIST)<etx>

FILE (MY)/#3152 START TRANSMISSION  
FILE "U/(MY)/DMPALL" START SENDING  
FILE (MY)/#3152 END TRANSMISSION  
FILE "U/(MY)/DMPALL" SENT

--> <sycom job #> AX.CONNECT<etx>

CONNECT INITIATED [028] Aa aA  
CONNECT COMPLETED

B1800/B1700 SMCS MARK IX.0.COD (08/20/79 13:40)  
(MY) LOGGED ON -SMCS- AT 09:36 08/22/79  
YOU ARE AT STATION 13 (VIRTUAL)  
News: This is Server1 SMCS, may I be of help?

--> <sycom job #> AX.WHAT<etx>

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; 2 FILES QUEUED;  
SENDING: (JH)/#665 (7901/15325);  
THIS STATION CONNECTED [028] Aa aA

USERCODE= MY, LOCAL HOST= USER1, REMOTE HOST= SERVER1

--> <sycom job #> AXZ COPY DMPALL AS ADMP;<etx>

SYSTEM/COPY =<job #> BOJ. PP=9, MP=9 TIME = 09:38:54.3  
"COPY 909" REMOVED  
"DMPALL" COPIED FROM SYSTEM DISK TO SYSTEM DISK AS ADMP  
SYSTEM/COPY =<job #> 1 FILE COPIED  
SYSTEM/COPY =<job #> EOJ. TIME = 09:41:33.6

--> <sycom job #> AX.DISCONNECT<etx>

STATION DISCONNECTED

--> <sycom job #> AX.FETCH ADMP<etx>

FILE ADMP START TRANSMISSION  
FILE "U/(MY)/ADMP" START RECEPTION  
FILE "U/(MY)/ADMP" RECEIVED  
FILE ADMP END TRANSMISSION

--> <sycom job #> AX.BYE<etx>

(MY) LOGGED OFF SYCOM AT 09:56:11.8 ON 22 AUG 79

-----//-----

### SYCOM PROGRAM SHUTDOWN

When SYCOM is running as a self-sustaining entity (without the SMCS), the ".EOJ" command provides the means by which an orderly shutdown can occur. Once this command has been entered, all subsequent commands entered are rejected. An appropriate "close message" is sent to the other SYCOM after the completion of all in-process file transfers. SYCOM then sends a "disconnect message" to the Network Controller and goes to "end-of-job". To initiate an orderly shutdown of SYCOM, the system operator must enter the following:

--> <sycom job #> AX.EOJ<etx>

The following message is displayed immediately on the User ODT:

2 SYCOM =<job #> "EOJ" IN PROGRESS. PLEASE SIGN OFF

No further processing requests are accepted. Any file transfers queued to process are allowed to complete. At the point in time when there is no longer any SYCOM activity and all users (including possible program input) are disconnected, SYCOM goes to "EOJ". The standard MCP "EOJ message" is displayed for this action:

SYCOM =<job #> EOJ. TIME = 16:21:55.1

When SYCOM is running in a self-sustaining environment and it is necessary to shutdown the system "immediately", then the ".STOP" command provides this means for termination. Upon entry of this command, all in-process activity is aborted. A "close message" is then sent to the other SYCOM, only if switched line, a "disconnect message" sent to the Network Controller, and the program goes to "end-of-job". The system operator enters the following to initiate the immediate shutdown of SYCOM:

```
--> <sycom job #> AX.STOP<etx>
```

The following messages are then displayed on the system ODT:

```
Z SYCOM =<job #> "STOP" IN PROGRESS
SYCOM =<job #> ECJ. TIME = 16:21:55.1
```

When SYCOM is running in conjunction with the SMCS, the "STOP" command entered to the SMCS in turn causes SYCOM to be terminated automatically. When this occurs, the following message will be displayed on the ODT:

```
SYCOM =<job #> DEATH IN FAMILY. TIME 16:21:55.1
```

A note of importance: If there are any requests queued for processing, they are retained if SYCOM shuts down under these circumstances.

#### PRINTING THE TRACE FILE

The disk TRACE file can be printed with the ".LIST" command.

#### REACTIONARY HINTS AND POINTERS

During the course of SYCOM operations, there are a few unusual system "conditions" which could occur. In order for the system operator to correctly respond to these conditions, there must be sufficient information available. That information is provided here in the form of problem descriptions, "hints", "pointers", and responses to these conditions.

A. When SYCOM appears to be "hung-up", enter "<job #>WY" at the system ODT. If the following message is displayed:

```
SYCOM =<job #> WAITING MAGNETIC TAPE <tape label>
```

then SYCOM has been automatically stopped waiting for the required magnetic tape to become available; at which time SYCOM operations will automatically resume.

This condition is resolved by the operator making a scratch magnetic tape available to the system.

Should the following message be displayed:

SMCS =<job #> WAITING QUEUE NOT FULL FOR WRITE

then SYCOM has been automatically stopped from "writing" any more messages into the Network Controller's input queue until it can successfully deliver at least one output message.

If this condition does not resolve itself after some period of time (2 - 5 minutes), the system must be brought down with the MCP "QC" command. (A system includes the SMCS, Network Controller, CANDE, and any other program associated with the Network Controller.)

Or if the following message is displayed; after SYCOM has been just executed:

SYCOM =<job #> EXECUTING

Then the problem lies in the fact SYCOM is attempting to establish a line connection to a remote system. It will continue in this cycle until either the connection has been established or the program DS'd. No input commands are entertained during this initial establishment cycle.

B. If SYCOM appears to be "stuck" during a file transfer, and the remote system is still "online", then either the "ONLINE" or "RESTART" commands may be tried.

1) The "ONLINE" command will break the connection and attempt to re-establish remote connection. If successful, SYCOM will automatically start sending the aborted file again.

2) The "RESTART" command, with no operand, will cause the same procedure to be invoked, except that the current PUT QUEUE will be thrown away. If however, an operand, either an implicit or relative schedule number is entered, then the system will start transmission of the specified file after evaluating the schedule number.

C. If the following message is displayed (by the MCP) on the system ODI during the initialization phase:

INSUFFICIENT DISK WHILE OPENING QUEUE-FILE "SYCOM"

It means that there is not sufficient space on disk to open the "SYCOM/LINK.G" queue. Affirmative action by the system operator would be to make available a sufficient amount of disk space (approximately 2000 segments).

## TERMINAL PROCEDURES

In this sub-section, basic terminal operating procedures are provided in the form of examples. For a SYCOM oriented system however, terminals may be owned by either SYCOM or the SMCS. Because of these possibilities and the associated implications, specific operating procedures are included for both "SYCOM controlled" and "SMCS controlled" terminals.

## TERMINAL FUNCTIONS

Terminal functions are those software features found in a TD820, and/or TD830 type Video Display Terminal (VDT) devices. These terminal features would include such things as; tab stops, lower/upper case, scrolling, etc.. These functions are fully described in the Supervisory Message Control System (SMCS) Software Operational Guide (see documentation reference).

## TERMINAL OPERATIONS (SYCOM CONTROLLED)

When a local terminal is under the direct control of SYCOM, it is also permanently "attached" to it. Therefore, a terminal user has immediate access to SYCOM functions. A terminal can however, be temporarily "attached" to either SYCOM or the SMCS in a Server System. Thru Server System's SMCS, the terminal can also be "attached" to an appropriate interactive application program in the Server System. The terminal user is not limited to just a few SMCS commands, a user may invoke any command which the SMCS has in its repertoire; with the exception of those restricted to input from the ODT.

### Example Interactive Terminal Session

Terminals at this point are attached to the User System's SYCOM. In order to utilize the SYCOM capabilities, a terminal user need only enter the desired SYCOM commands. Among these SYCOM commands are ".CONNECT" and ".DISCONNECT". These are used to gain access to any appropriate interactive application program in the Server System. The examples which follow explore all possible terminal connections as detailed in "Message Path Assignments".

In the examples below, terminal input is identified by the flag "-->". All other lines are reply messages from the program to which the terminal is currently inputting to. First some basic local SYCOM commands are shown. All SYCOM commands are entered directly as exemplified below:

--> .WHAT

(User SYCOM)

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE  
THIS STATION NOT CONNECTED  
LOCAL HOST = USER1, REMOTE HOST = SERVER1

--> .PUT DMPALL (User SYCOM)

FILE DMPALL START TRANSMISSION

--> .WHAT (User SYCOM)

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE;  
0 FILES QUEUED; SENDING DMPALL (47/85)  
THIS STATION NOT CONNECTED  
LOCAL HOST = USER1, REMOTE HOST = SERVER1  
FILE DMPALL END TRANSMISSION

--> .CONNECT (User SYCOM)

CONNECT INITIATED [029] Bb bB  
CONNECT COMPLETED to HOST = <hostname>

--> .WHAT (User SYCOM)

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE  
THIS STATION CONNECTED [029] Bb bB  
LOCAL HOST = USER1, REMOTE HOST = SERVER1

--> ML;MX (Server SMCS)

MIX LIMIT=63  
SERVER/NC =<job #> PP=15, MP=15  
SMCS =<job #> PP=13, MP=14  
SERVER/SYCOM =<job #> PP=12, MP=12  
RD =<job #> PP=11, MP=11  
END MX

It is necessary that the terminal user be aware of what "signal character" the Server System's SMCS is currently using. To find out, the following command is entered to query that character:

--> SIGNAL (Server SMCS)

CURRENT SIGNAL CHARACTER = \$.

Now the terminal user can "sign on" to an application program in the Server System; and alternately enter commands to the Server System's SMCS. A series of following interactions exemplify this concept: (The application program "SET/TABS" used in the example is assumed to "set" specified tab stops in the user's terminal.)

--> SIGN ON SET/TABS (Server SMCS)

SIGNED ON TO "SET/TABS", SIGNAL = \$

--> \$ML;MX (Server SMCS)

MIX LIMIT=63  
SERVER/NC =<job #> PP=15, MP=15  
SMCS =<job #> PP=13, MP=13  
SERVER/SYCOM =<job #> PP=12, MP=12  
SET/TABS =<job #> PP=12, MP=12  
RC =<job #> PP=11, MP=11  
END MX

--> 10,20,30,40,50,60,70 (Server AP)

(tab stops are now set)  
REMOTE FILE CLOSED BY "SET/TABS".

--> EX DMPALL;FI OUTP.FILE INV 2 (Server SMCS)

DMPALL =<job #> BOJ. PP=9, MP=9 TIME = 11:35:21.8  
% DMPALL =<job #> ENTER SPECS  
DMPALL =<job #> ACCEPT.

--> <dmpall job #> AX COPY DSKLST SMCS/TRACE

DMPALL =<job #> "OUTPUT" = "BACKUP.PRT/906"  
DMPALL =<job #> "BACKUP.PRT/906" RELEASED  
% DMPALL =<job #> ENTER SPECS  
DMPALL =<job #> ACCEPT.

--> <dmpall job #> AX

DMPALL =<job #> EOJ. TIME = 11:36:58.1

At the point in time when a user is finished entering commands to SYCOM and wishes to terminate the session, the following command is entered; however, the terminal is still "attached" to SYCOM.

--> .DISCONNECT (User SYCOM)

STATION DISCONNECTED

-----//---

### Terminal Operations (SMCS Controlled)

When a terminal is permanently attached to, and owned by, the SMCS, the terminal user must gain access to SYCOM functions thru the use of explicit SMCS commands. These commands are exemplified below. By gaining access to SYCOM, the SMCS will temporarily "attach" the requesting terminal to SYCOM for the duration of a session. However, a terminal user is not limited to just a few SMCS commands, a user may invoke any command which the SMCS has in its repertoire; with the exception of those restricted to input from the ODT.

#### Example 1: Interactive Session

The terminal at this point is attached to the User System's SMCS. In order to gain access to those rifty SYCOM capabilities, the terminal user must make use of the SMCS "SIGN ON" and "SIGN OFF" commands. The sample session which follows, shows the terminal user just how easy it is to "get on" and "get off" SYCOM: (input from the terminal is identified by the flag "-->".)

```
--> SIGN ON SYCOM (User SMCS)
```

```
SIGNED ON TO "SYCOM", SIGNAL = *
```

When the "signed-on" message is returned to the terminal, that terminal is then attached to SYCOM. Therefore, SYCOM commands are entered directly as exemplified below:

```
--> .WHAT (User SYCOM)
```

```
SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE
THIS STATION NOT CONNECTED
LOCAL HOST = USER1, REMOTE HOST = SERVER1
```

```
--> .PUT DMPALL (User SYCOM)
```

```
FILE DMPALL START TRANSMISSION
```

```
--> .WHAT (User SYCOM)
```

```
SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE;
0 FILES QUEUED; SENDING DMPALL (47/85)
THIS STATION NOT CONNECTED
LOCAL HOST = USER1, REMOTE HOST = SERVER1
FILE DMPALL END TRANSMISSION
```

```
--> .CONNECT (User SYCOM)
```

```
CONNECT INITIATED [028] Aa aA
CONNECT COMPLETED to HOST = <hostname>
```

--> .WHAT (User SYCOM)

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE  
THIS STATION CONNECTED [C28] Aa aA  
LOCAL HOST = USER1, REMOTE HOST = SERVER1

--> \*REPORT > (User SMCS)

>R1 (ODT): ON JOB #15 SYCOM  
1 STATION REPORTED AT 11:42 08/22/79

--> MLI;MX (Server SMCS)

MIX LIMIT=63  
SERVER/NC =<job #> PP=15, MP=15  
SMCS =<job #> PP=13, MP=14  
SERVER/SYCOM =<job #> PP=12, MP=12  
RD =<job #> PP=11, MP=11  
END MX

It is necessary that the SMCS in a Server System have a "signal character" that is different than that of the SMCS in the User System. To ensure this, the following command is entered to query that signal character: (If it "is not" different, then change it; "SIGNAL \$".)

--> SIGNAL (Server SMCS)

CURRENT SIGNAL CHARACTER = \$

Now the terminal user can "sign on" to an application program in the Server System; and alternately enter commands to the Server System's SMCS. A series of following interactions exemplify this concept: (The application program "SET/TABS" used in the example is assumed to "set" specified tab stops in the user's terminal.)

--> SIGN ON SET/TABS (Server SMCS)

SIGNED ON TO "SET/TABS", SIGNAL = \$

--> \$MLI;MX (Server SMCS)

MIX LIMIT=63  
SERVER/NC =<job #> PP=15, MP=15  
SMCS =<job #> PP=13, MP=13  
SERVER/SYCOM =<job #> PP=12, MP=12  
RD =<job #> PP=11, MP=11  
SET/TABS =<job #> PP=4, MP=4  
END MX

--> \*REPORT > (User SMCS)

>R1 (ODT): ON JOB #15 SYCOM  
1 STATION REPORTED AT 11:44 08/22/79

--> 10,20,30,40,50,60,70 (Server AP)

(tab stops are now set)  
REMOTE FILE CLOSED BY "SET/TABS".

--> .DEBUG (User SYCOM)

DEBUG OFF

--> .DISCONNECT (User SYCOM)

STATION DISCONNECTED

At the point in time when a user is finished entering commands to SYCOM and wishes to terminate the session, the following command is entered: (This is a SMCS command.)

--> \*SIGN OFF (User SMCS)

The response from the SMCS will be as follows:

SIGNED OFF "SYCOM"

At this point, the terminal has been "detached" from SYCOM and is now "re-attached" to the SMCS.

### Example 2: Interactive Session, System Strings

This example of a typical interactive session whereby the user on one system communicates through several other systems in a string of systems to direct the end system in specific functions. To do this, example 2 utilizes the SMCS "PASS" command exclusively. It provides the means to establish communication links to an end system without having to "sign-on" to each system in the link.

At this point the terminal is attached to the User System's SMCS. In order to gain access to the SYCOM system-string capabilities, the terminal user can either make use of the SMCS's "SIGN ON" and "SIGN OFF" commands or the "PASS" command. For this example, the PASS command will be utilized.

For this example of accessing system-strings, the distributed system complex we all know as the HUB/Slave systems was chosen. Therefore, each step of the following procedure can be compared against the string-system schematic on the following page. With this combination, a user can go as far into the system-strings as desired.

In the following procedure, example terminal input is shown in "lower case"; upper or lower case is allowable. Terminal input is also flagged with the "--->" reference identifier.

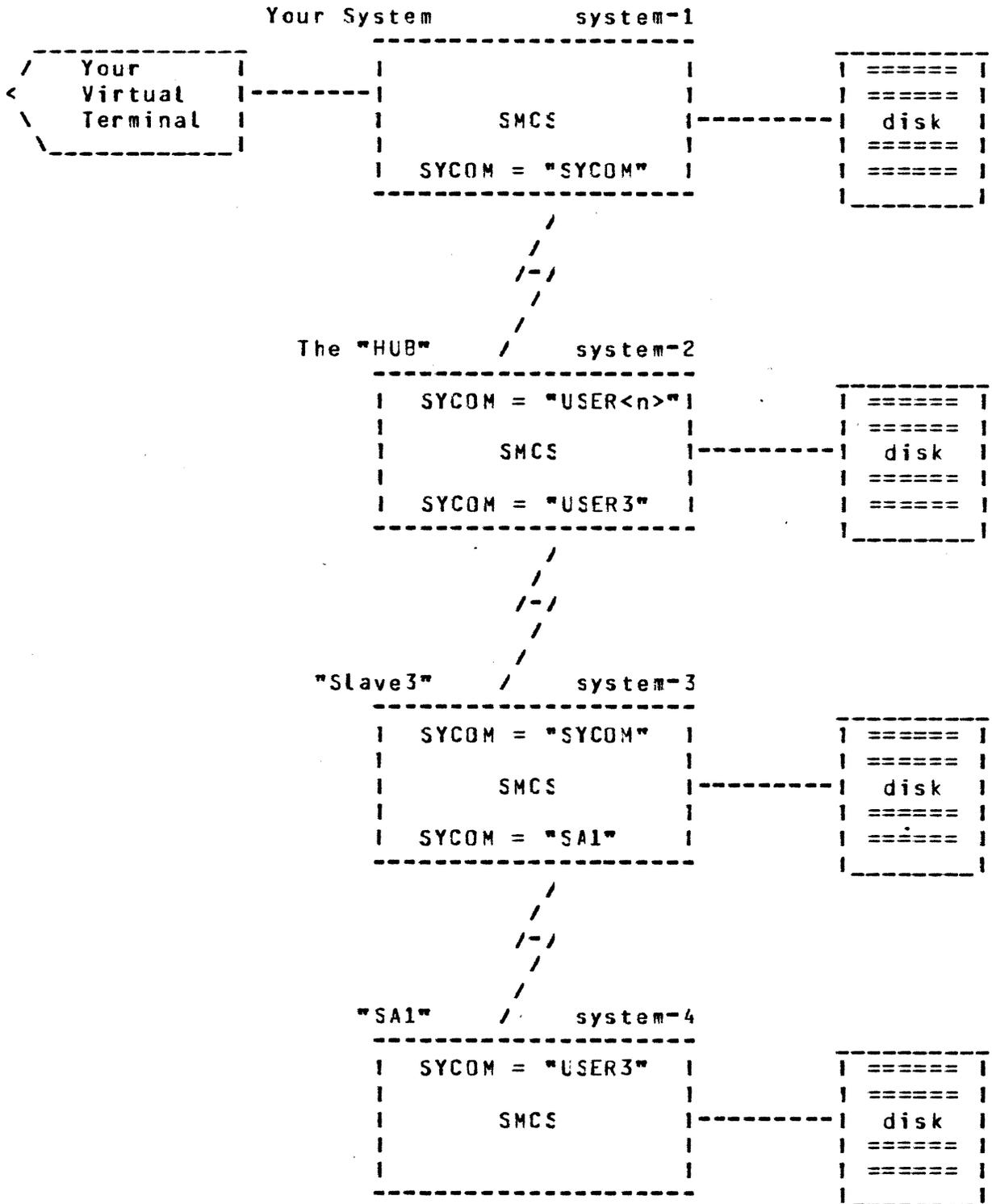


Figure V-1: Multiple System Message Path Schematic

We will begin by logging-on to the user's SMCS. This establishes a base connection, however, it is not a requirement.

```
--> id "sycom user" -----> (SMCS; sys 1)
```

```
B1800/B1700 SMCS MARK IX.0.00D (08/20/79 13:40)
"sycom user" LOGGED ON -SMCS- AT 08:46 08/30/79
YOU ARE AT STATION 1 (TDC832)
```

If "US A/B" is entered then user A with password B is logged on to the local SMCS. All subsequent PASSES to SYCOM will automatically include this usercode/password pair. Thus, if the user is declared to be a workstation, file security is enforced.

When the "logged-on" message is returned to the terminal, that terminal is then attached to the SMCS. Hereafter, SYCOM commands are entered via a series of "PASS" messages as follows:

```
--> p sycom .connect -----> (SYCOM; sys 1)
```

```
CONNECT INITIATED [028] Aa aA
CONNECT COMPLETED
```

```
--> p sycom .signal 1 -----> (SYCOM; sys 1)
```

```
STATION SIGNAL IS "1"
```

```
--> p sycom lwhat -----> (SYCOM; sys 1)
```

```
SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; 1 FILE QUEUED;
SENDING: (7901/15325);
THIS STATION CONNECTED [028] Aa aA
LOCAL HOST = SL2, REMOTE HOST = HUB
```

One will not see filenames since it is a usercoded file, unless logged on the SMCS as usre (JH). (See above)

```
--> p sycom system id -----> (SMCS; sys 2)
```

```
SYSTEM-ID: THE.HUB
```

```

Current terminal PASS message paths: #

VDT input -----> <xx> SMCS sys 1 #
|----> p sycom l<xx> SYCOM sys 1 #
|----> p sycom <xx> SMCS sys 2 #
#####
```

```
--> p sycom ml;mx -----> (SMCS; sys 2)

MIX LIMIT=63
HUB/NC.JUL26 =<job #> PP=15, MP=15
SMCS =<job #> PP=13, MP=14
(USER1)/SYCOM =<job #> PP=12, MP=12
(USER3)/SYCOM =<job #> PP=12, MP=12
END MX
```

At this point, as shown in the message path block above, the user has established a communication link to a second system. Input can be sent the SMCS, MCP, or SYCOM in either of the two systems.

If a connection is desired to a third system, then continue with be entering the following command:

```
--> p sycom p user3 .connect -----> (SYCOM; sys 2)

CONNECT INITIATED [031] Aa aA
CONNECT COMPLETED to HOST = <hostname>

--> p sycom p user3 .signal 2 -----> (SYCOM; sys 2)

STATION SIGNAL IS "2"

--> p sycom p user3 2what -----> (SYCOM; sys 2)

(USER3)/SYCOM IX.0.3 (08 AUG 79 15:19) ONLINE; IDLE.
THIS STATION CONNECTED [031] Aa aA
LOCAL HOST = HUB, REMOTE HOST = SLAVE3

--> p sycom p user3 system -----> (SMCS; sys 3)

SYSTEM-ID: SLAVE3
SYSTEM-TRANSLATE: ON
SYSTEM-VERSION: 81800/B1700 SMCS MARK IX.0.00D
```

```
#####
Current terminal PASS message paths:
#
VDT input -----> <xx> SMCS sys 1
|-----> p sycom l<xx> SYCOM sys 1
|-----> p sycom <xx> SMCS sys 2
|-----> p sycom p user3 2<xx> SYCOM sys 2
|-----> p sycom p user3 <xx> SMCS sys 3
#####
```

The user now has access, via SYCOM connections, to a third system in the string of systems; as pictured in the message path block above. If it is desirable to extend these connections to a fourth system, then enter the following command string:

--> p sycom p user3 p sa1 .connect -----> (SYCOM; sys 3)

CONNECT INITIATED [036] Aa aA  
CONNECT COMPLETED to HOST = <hostname>

--> p sycom p user3 p sa1 .signal 3 -----> (SYCOM; sys 3)

STATION SIGNAL IS "3"

--> p sycom p user3 p sa1 3what -----> (SYCOM; sys 3)

SA1 IX.0.3 (08 AUG 79 15:19) ONLINE; IDLE.  
THIS STATION CONNECTED [036] Aa aA  
LOCAL HOST = SLAVE3, REMOTE HOST = SA1

```

Current terminal PASS message paths: #

VDT input -----> <xx> SMCS sys 1 #
|---> p sycom 1<xx> SYCOM sys 1 #
|---> p sycom <xx> SMCS sys 2 #
|---> p sycom p user3 2<xx> SYCOM sys 2 #
|---> p sycom p user3 <xx> SMCS sys 3 #
|---> p sycom p user3 p sa1 3<xx> . SYCOM sys 3 #
|---> p sycom p user3 p sa1 <xx> . SMCS sys 4 #
#####
```

At this point, the user has connections all the way thru to the SMCS in system "SA1". Commands can be given to the SMCS or MCP in that system or any system along the way in the message path connections.

--> p sycom p user3 p sa1 system -----> (SMCS; sys 4)

SYSTEM-ID: SA1  
SYSTEM-TRANSLATE: ON  
SYSTEM-VERSION: B1800/B1700 SMCS MARK IX.0.00D

--> p sycom p user3 p sa1 ml;#x -----> (SMCS; sys 4)

MIX LIMIT=63  
SA1/NC =<job #> PP=15, MP=15  
SMCS =<job #> PP=14, MP=14  
(USER3)/SYCOM =<job #> PP=13, MP=13  
CANDE =<job #> PP=13, MP=13  
RD =<job #> PP=13, MP=13  
END MX

With the current inter-system connections, a terminal user can "sign-on" to programs in either of the four systems and/or issue SMCS and MCP commands in those systems.

In the following example, a compile is initiated in the forth system, without usercode security being enforced:

```
--> p sycom p user3 p sa1 co px sdi;fi cards nam xs dsk;
```

```
PX =<job #> BOJ. PP=4, MP=4 TIME = 09:00:20.6
```

```
PX =<job #> EOJ. TIME = 09:02:08.8
```

After all transactions with the SA1 system have been completed, all connects must be "disconnected". This must be accomplished with the following command procedure: (By giving a "bye" command to each SYCOM, the "station signal character" is restored to "." and the virtual station is automatically "disconnected".)

```
--> p sycom p user3 p sa1 3bye -----> (SYCOM; sys 3)
```

```
STATION DISCONNECTED
```

```
USER SY LOGGED OFF TO SA1 AT 09:01:21.3 ON 30 AUG 79
```

```
--> p sycom p user3 2bye -----> (SYCOM; sys 2)
```

```
STATION DISCONNECTED
```

```
USER SY LOGGED OFF TO (USER3)/SYCOM AT 09:02:44.1 ON
```

```
--> p sycom 1bye -----> (SYCOM; sys 1)
```

```
STATION DISCONNECTED
```

```
USER SY LOGGED OFF TO SYCOM AT 09:03:38.1 ON 30 AUG 79
```

```
--> p sycom what -----> (SYCOM; sys 1)
```

```
SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE
```

```
THIS STATION NOT CONNECTED
```

```
LOCAL HOST = SL2, REMOTE HOST = HUB
```

At this point, control of the terminal has reverted to the SMCS.

### Example 3: Interactive Session, System Strings

This is other example of a typical interactive session when the user on a local system communicates through several other systems in a "string of systems" to direct an "end system" to perform a series of specific functions; of the user's discretion.

In this example, use of the SMCS "SIGN ON" command is utilized to provide the means to establish the communication links to an "end system"; Signing-on" to each system in the link. By doing this, a user is assured that tasks can be directed to "any" system in the string and have the appropriate security applied as needed.

At this point the terminal is attached to the User System's SMCS. In order to gain access to the SYCOM system-string capabilities, the terminal user will make use of the SMCS's "SIGN ON" command.

For this example of accessing system-strings, the distributed system complex we all know as the HUB/Slave systems was chosen. Therefore, each step of the following procedure can be compared against the string-system schematic (Figure V-1). With this combination, a user can go as far into the system-strings as desired.

In the following procedure, example terminal input is shown in "lower case" (upper or lower case is allowable). Terminal input is also flagged with the "-->" reference identifier.

Refer to Figure V-1 for the string system message path schematic.

The user begins by "logging-on" to the local SMCS. This usercode and password will be used by both SYCOM and SMCS to automatically log the user on to each system in the link (SYCOM/SMCS):

```
--> us sy/user -----> (SMCS; sys 1)

B1800/B1700 SMCS MARK IX.0.00D (08/20/79 13:40)
(SY) LOGGED ON -SMCS- AT 09:13 08/30/79
YOU ARE AT STATION 1 (TD832)
```

The terminal is still attached to the SMCS in system 1. To make an attachment to SYCOM, enter the following "SIGN ON" command:

```
--> on sycom -----> (SMCS; sys 1)

SIGNED ON TO "SYCOM", SIGNAL = *
B1700/B1800 SYCOM IX.0.2 (30 JUL 79 14:53)
USER SY LOGGED ON TO SYCOM AT 09:13:38.1 ON 30 AUG 79

--> *make signal a -----> (SMCS; sys 1)

STATION <name> SIGNAL CHANGED TO a

--> .connect -----> (SYCOM; sys 1)

CONNECT INITIATED [028] Aa aA
CONNECT COMPLETED to HOST = <hostname>
B1800/B1700 SMCS MARK IX.0.00D (08/20/79 13:40)
(SY) LOGGED ON -SMCS- AT 09:15 08/30/79
YOU ARE AT STATION 21 (VIRTUAL)

--> .signal 1 -----> (SYCOM; sys 1)

STATION SIGNAL IS "1"

--> 1 what -----> (SYCOM; sys 1)

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE
THIS STATION CONNECTED [028] Aa aA
USERCODE = SY, LOCAL HOST = SL2, REMOTE HOST = HUB
```

Either one of the following maybe used in place of the above statements.

ON SYCOM .CONNECT  
or  
ON SYCOM. SIGNAL 1  
1CONNECT

```

Current terminal input message path assignments #

VDT input -----> 2<xx> . . . SMCS sys 1 #
|-----> 1<xx> . . . SYCOM sys 1 #
|-----> <xx> . . . SMCS sys 2 #
#####
```

```
--> system -----> (SMCS; sys 2)

SYSTEM-ID: THE.HUB
SYSTEM-TRANSLATE: ON
SYSTEM-VERSION: B1800/B1700 SMCS MARK IX.0.00D
```

Now the terminal user can establish a connection to any other of the systems in the string. At this point we can make connection to the third system (slave 3) by entering the following command:

```
--> on user3 -----> (SMCS; sys 2)

SIGNED ON TO "SYCOM", SIGNAL = <?>
B1700/B1800 SYCOM IX.0.3 (08 AUG 79 15:19)
USER SY LOGGED ON TO (USER3)/SYCOM AT 09:16:44.1 ON
```

It is necessary for the SMCS's "signal character" in System-2 be different than that of the SMCS in the System-1. If the "SIGNED ON TO "SYCOM"" message returned for the last sign-on is other than a "2", then skip the next input command; otherwise, change the "SMCS's signal character" in System-2 as follows:

```
--> <current signal> make signal $ -----> (SMCS; sys 2)

STATION <name> SIGNAL CHANGED TO $

--> .connect -----> (SYCOM; sys 2)

CONNECT INITIATED [031] #a aA
CONNECT COMPLETED to HOST = <hostname>
B1800/B1700 SMCS MARK IX.0.00D (08/20/79 13:40)
(SY) LOGGED ON -SMCS- AT 09:17 08/30/79
YOU ARE AT STATION 41 (VIRTUAL)

--> .signal 2 -----> (SYCOM; sys 2)

STATION SIGNAL IS "2"
```

```

Current terminal input message path assignments #

VDT input -----> a<xx> . . . SMCS sys 1 #
|-----> 1<xx> . . . SYCOM sys 1 #
|-----> $<xx> . . . SMCS sys 2 #
|-----> 2<xx> . . . SYCOM sys 2 #
|-----> <xx> . . . SMCS sys 3 #
#####
```

--> system -----> (SMCS; sys 3)

SYSTEM-ID: SLAVE3  
SYSTEM-TRANSLATE: ON  
SYSTEM-VERSION: B1800/B1700 SMCS MARK IX.0.00D

--> on sa1 -----> (SMCS; sys 3)

SIGNED ON TO "SYCOM", SIGNAL = \*  
B1700/B1800 SYCOM IX.0.3 (08 AUG 79 15:19)  
USER SY LOGGED ON TO SA1 AT 09:18:59.3 ON 30 AUG 79

It is necessary for the SMCS's "signal character" in System-3 be different than those for system's 1 and 2. If the "SIGNED ON TO "SYCOM"" message returned for the last sign-on is other than a "a" or "\$", then skip the next input command; otherwise, change the "SMCS's station signal character" in System-3 as follows:

--> <current signal> make signal # -----> (SMCS; sys 3)

STATION <name> SIGNAL CHANGED TO #

--> .connect -----> (SYCOM; sys 3)

CONNECT INITIATED [028] Aa aA  
CONNECT COMPLETED  
B1800/B1700 SMCS MARK IX.0.00D (08/20/79 13:40)  
(SY) LOGGED ON -SMCS- AT 09:20 08/30/79  
YOU ARE AT STATION 41 (VIRTUAL)

--> .signal 3 -----> (SYCOM; sys 3)

STATION SIGNAL IS "3"

At this point, the user is now connected to, and logged-on, the SMCS in the "SA1" system.

```

Current terminal input message path assignments #

VDT input -----> a<xx> . . . SMCS sys 1 #
|-----> 1<xx> . . . SYCOM sys 1 #
|-----> $<xx> . . . SMCS sys 2 #
|-----> 2<xx> . . . SYCOM sys 2 #
|-----> #<xx> . . . SMCS sys 3 #
|-----> 3<xx> . . . SYCOM sys 3 #
|-----> <xx> . . . SMCS sys 4 #
#####
```

--> system -----> (SMCS; sys 4)

SYSTEM-ID: SA1  
SYSTEM-TRANSLATE: ON  
SYSTEM-VERSION: B1800/B1700 SMCS MARK IX.0.00D

The connections exist to allow input messages from your terminal to be sent to the SMCS in the "SA1" system. At this point, one is free to "PASS" to programs in that system and/or give commands to the SMCS/MCP there:

--> co px sdi;fi cards ram (sy)/xs dsk; ---> (SMCS; sys 4)

(SY) PX =<job #> BOJ. PP=4, MP=4 TIME = 09:11:17.6  
(SY) PX =<job #> EOJ. TIME = 09:12:59.8

--> a system id -----> (SMCS; sys 1)

SYSTEM-ID: SL2

--> \$ system id -----> (SMCS; sys 2)

SYSTEM-ID: THE.HUB

--> # system id -----> (SMCS; sys 3)

SYSTEM-ID: SL3

--> 1 what -----> (SYCOM; sys 1)

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE  
THIS STATION CONNECTED [C28] Aa aA  
USERCCDE = SY, LOCAL HOST = SL2; REMOTE HOST = HUB

--> 2 what -----> (SYCOM; sys 2)

(USER3)/SYCOM IX.0.3 (08 AUG 79 15:19) ONLINE; IDLE.  
THIS STATION CONNECTED [C41] Aa aA  
USERCCDE = SY, LOCAL HOST = HUB, REMOTE HOST = SLAVE3

--> 3 what -----> (SYCOM; sys 3)

SA1 IX.0.3 (08 AUG 79 15:19) ONLINE; IDLE.  
THIS STATION CONNECTED [040] Aa aA  
USERCODE = SY, LOCAL HOST = SL3, REMOTE HOST = SA1

--> ex dmpall;fi outp.file inv 2;pr 11;

(SY) DMPALL =<job #> BOJ. PP=11, MP=11 TIME = 09:22:1  
% DMPALL =<job #> ENTER SPECS.  
(SY) DMPALL =<job #> ACCEPT.

--> <job #>AX COPY DSKLST SMCS/TRACE;

% DMPALL =<job #> ENTER SPECS.  
(SY) DMPALL =<job #> ACCEPT.

--> <job #>AX

(SY) DMPALL =<job #> EOJ. 09:23:11.2

After all transactions with the SA1 system have been completed, all connects must be "disconnected". This can be accomplished with the following list of commands: (Again, by giving a "bye" command to each SYCOM in the chain, the "station signal character" is restored to "." and the virtual station is automatically "disconnected".)

--> 3bye -----> (SYCOM; sys 3)

STATION DISCONNECTED  
USER (SY) LOGGED OFF SA1 AT 09:23:33.1 ON 30 AUG 79  
DETACHED FROM "SA1"

--> 2bye -----> (SYCOM; sys 2)

STATION DISCONNECTED  
USER (SY) LOGGED OFF (USER3)/SYCOM AT 09:33:23.7 ON  
DETACHED FROM "USER3"

--> 1disconnect -----> (SYCOM; sys 1)

STATION DISCONNECTED

--> what -----> (SYCOM; sys 1)

SYCOM IX.0.2 (30 JUL 79 14:53) ONLINE; IDLE  
THIS STATION NOT CONNECTED  
USERCODE = SY, LOCAL HOST = SL2, REMOTE HOST = HUB

--> bye -----> (SMCS; sys 1)

(SY) LOGGED OFF SYCOM AT 09:24:09.1 ON 30 AUG 79  
DETACHED FROM "SYCOM"

BURROLGHS CORPORATION  
COMPUTER SYSTEMS GROUP  
SANTA BARBARA PLANT

7-21  
COMPANY CONFIDENTIAL  
B1800/B1700 SYCOM  
P. S. 2219 0458 (A)

At this point, control of the terminal has reverted to the SMCS.

The \*OFF command may be used instead of the above list of disconnect commands. The \*OFF will disconnect one completely where as the above disconnect commands must be done one input message at a time.

## GENERATING NETWORK CONTROLLERS

### GENERAL

This section is provided as a user oriented guideline to the generation of a Network Controller for the SYCOM system. The controller is generated by the Network Definition Language (NDL) compiler using a prepared set of NDL specifications. The system user is required to determine these NDL specifications according to the system configuration and environment under which the Network Controller is to function.

The following guidelines are offered for a target system that is to provide the full range of SYCOM services including terminals. They are intended to aid the user in generating a Network Controller with correct functional attributes. For a complete description of the NDL specifications and definitions, refer to the "Network Definition Language reference manual" (associated documentation).

### DETERMINING THE SPECIFICATIONS

The following sample Network Controller NDL specifications are presented in a sequence similar to that of the NDL reference manual. This allows the user a logical reference for evaluating the sample NDL specifications for application to the installation requirements. They identify suggested parameters which produce an operational Network Controller program for the SYCOM system configuration as specified.

### OPERATIONAL PROCEDURES

The following procedures provide the methods by which the user prepares the input to the NDL compilation process, generates a Network Controller, and induces modifications into the resident NDL compiler, if required.

### PREPARATION OF INPUT

The NDL "source" input file can be prepared in the form of either a card file or a NDL type disk file. The disk file could be created and maintained using CANDE or another text editor of similar capabilities.

The "Sample NDL Parameters File" is furnished for the purpose of creating this file with the most amount of accuracy in the least amount of time. Any adjustments to this source file for number

of lines, terminals, and other such attributes is the responsibility of the user which is attuned to the system configuration and its operational environment.

#### COMPILATION PROCESS

Prior to initiating the NDL compilation process, the user should ensure the presence of a "10.0 NDL/LIBRARY" file on disk, of a 25 Oct 1980 or later vintage). Then, there are several methods by which the NDL compilation (generative) process can be invoked. Following are two of these methods:

##### Card Input:

Following is a primitive example of a NDL source card input deck: (The <I> in the example below represents an "invalid" character as required by the MCP.)

```
<I> CO USER/NC NDL LI
<I> DATA CARDS
```

```
NDL source images
NDL library calls
```

```
<I> END
```

##### Disk Input:

This method is used to initiate a NDL compiler process when the source input file resides on disk. Enter the following MCP command using the system ODI:

```
CO USER/NC NDL LI;FI CARDS NAM USERNC/SOURCE DSK DEF
```

From this point on, the compilation procedures are the same. When the MCP executes the NDL compiler, a normal "BOJ" message is displayed. Upon successful completion of the compile, the normal "EOJ" message is displayed. However, if an error message such as the one below is displayed:

```
NDL : USER/NC =<job #> --EVALUATION/PROGRAM POINTER STACK
OVERFLW DS OR DP
```

Then the "NDL Compiler" must be modified to increase its "name/value stack" size.

#### PRINT THE USER/NCNIF FILE

Once the compilation/generation of the Network Controller is complete, it is suggested that the "Network Information File" be printed. This is accomplished by executing the following program; and providing the correct file name (if different than the one suggested):

EX NIF/ANALYZER; FI NIF NAM USER/NCNIF<etx>

NIF/ANALYZER =<job #> BOJ. PR=4 TIME = 11:05:27.1  
NIF/ANALYZER =<job #> EOJ. TIME = 11:07:12.9

**SAMPLE NDL SOURCE FILE (LOCAL USER SYSTEM)**

Following is a sample NDL source file to be used as input to the NDL compiler for the generation of a Network Controller oriented to a User System. This example provides a Network Controller which services a local site of 3 terminals, 2 virtual terminals, and a inter-system connection. The user must tailor this sample Network Controller specifications to the needs and desires of the installation (e.g., more/less local terminals, more/less virtual terminals, etc).

This sample NDL input file is comprised of both NDL statements and parameter specifications. The resulting Network Controller will provide the correct functional attributes necessary for the operation of the SYCOM system; in testing mode.

```
$ CONTROL
$ LIBINFO
$ NSSIZE 20 % number of entries; for IOLOG
$ VSSIZE 15000C % number of bits; for IOLOG
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X X
X Declaration Section X
X X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
DECLARATION:
 NIF = "USER"/"NCNIF". % NIF file name; user option
 MAX FILES = 16.
 MAX BUFFERS = 2.
 MAX MESSAGES = 20.
```

```
$ LIBRARY SYCOMDEC

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X X
X Requests and Controls X
X X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
$ LIBRARY SYCOMREQ
$ LIBRARY CANDEPOLTD %**
$ LIBRARY CANDEFSLTD %**
$ LIBRARY SYCOMCTL
$ LIBRARY CANDETCTL %**
$ LIBRARY SYCOMREQ
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X Network Configuration X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X Terminal Section X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

TERMINAL DEFAULT SYCOMDEFAULT:  
REQUEST = SYCOMREQ:RECEIVE, SYCOMREQ:TRANSMIT.  
TRANSMISSION = 1.  
ADDRESS = 2.                   Z Number of polling address characters  
BUFFERSIZE = 3840.            Z 2 x 1920 bytes  
TYPE = 62.                    Z B1800

TERMINAL SYCOMTERMINAL:  
DEFAULT = SYCOMDEFAULT.

TERMINAL DEFAULT TDFLT:                   Z\*\*  
REQUEST = CANDEPOLTD:RECEIVE,        Z\*\*  
          CANDEFSLTD:TRANSMIT.       Z\*\*  
TRANSMISSION = 0.                    Z\*\*  
ADDRESS = 2.                         Z\*\*  
TYPE = 44.                         Z\*\* TD820  
BUFFERSIZE = 2250.                   Z\*\*

TERMINAL TD824:                         Z\*\*  
DEFAULT = TDFLT.                     Z\*\*

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X Station Section X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

STATION DEFAULT SYCOMDEF:  
FREQUENCY = 250, 250.            Z     MCS Controller  
MYUSE = INPUT, OUTPUT.  
TERMINAL = SYCOMTERMINAL.  
RETRY = 25.

STATION DEFAULT TD8DF:                   Z\*\*  
FREQUENCY = 249, 249.            Z\*\*    Non-Remote ODT  
MYUSE = INPUT, OUTPUT.            Z\*\*  
TERMINAL = TD824.                 Z\*\*  
RETRY = 4.                         Z\*\*

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

%% If any RJE station are required for "RJE/HOST", then they  
%% must be declared LSN's 1 - 4; no exceptions.....

```
STATION CONTROLST1: % -----
 DEFAULT = SYCOMDEF. % lsn 001 |
 ADDRESS = "00". % |
 % |
 % |
STATION SPOST2: % |
 DEFAULT = SYCOMDEF. % lsn 002 |
 ADDRESS = "01". % | Must be grouped.
 % |
 % |
STATION CARDST3: % > -----
 DEFAULT = SYCOMDEF. % lsn 003 |
 ADDRESS = "02". % | Don't have to be
 % | first stations
 % | defined.
STATION PRINTERST4: % |
 DEFAULT = SYCOMDEF. % lsn 004 |
 ADDRESS = "03". % |
 % |
 % |
STATION FILEST5: % |
 DEFAULT = SYCOMDEF. % lsn 005 |
 ADDRESS = "04". % -----
```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```
STATION USERT1: % lsn 006 * Virtual terminal
 DEFAULT = SYCOMDEF. % (ODT/terminals)
 ADDRESS = "Aa", "aA". % |
 % |
STATION USERT2: % lsn 007 * Virtual Terminal
 DEFAULT = SYCOMDEF. % (Program)
 ADDRESS = "Pa", "aP". % |
```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```
STATION R1: %** -----
 DEFAULT = TC8DF. %** lsn 008 |
 ADDRESS = "R1". %** |
 FREQUENCY = 249, 250. %** Remote ODT |
 %** |
STATION R2: %** lsn 009 |
 DEFAULT = TC8DF. %** > Local terminal
 ADDRESS = "R2". %** |
 FREQUENCY = 249, 250. %** Remote ODT |
 %** |
STATION R3: %** lsn 010 |
 DEFAULT = TC8DF. %** |
 ADDRESS = "R3". %** |
 FREQUENCY = 249, 250. %** Remote ODT |
 %** -----
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X Line Section X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

LINE DIRECT01:

```
CONTROL = SYCOMCTL.
STATION = CONTROLST1, SPOST2, CARDST3, PRINTERST4, FILEST5,
 USERT1, USERT2.
ADDRESS = 1:0:6. X (line, adapter, port; system)
```

LINE DEFAULT LINEDFLT:

```
CONTROL = CANDETDCTL. X**
AUTCPOLL = 15. X** 5 x max stations/lines
```

LINE MULTI04:

```
DEFAULT = LINEDFLT. X** adapter 4
STATION = R1, R2, R3. X**
ADDRESS = 1:0:4. X** (line, adapter, port; system)
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

FILE LINK:

```
FAMILY = CONTROLST1, SPOST2, CARDST3, PRINTERST4, FILEST5,
 USERT1. X for SYCOM
 X rsn's 001, 002, 003, 004,
 X 005, 006
```

FILE RFUSER:

```
FAMILY = USERT2. X For program-to-program I/O.
 X rsn 001
```

FILE TERMINALS:

```
X** This file is named "TERMINALS"
X** when SYCOM is run without the
X** SMCS. It must be named
X** "MCSREMOTE" when SYCOM is run
X** subordinate to the SMCS..
FAMILY = R1, R2, R3. X** rsn's 001, 002, 003
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

FINI.

-----//---

Note: If the User System does not have "local terminals" then all lines in the above example with comments indicated with "X\*\*" can be omitted from the source file.

SAMPLE NDL SOURCE FILE (REMOTE SERVER SYSTEM)

Following is a sample NDL source file to be used as input to the NDL compiler for the generation of a Network Controller oriented to a Server System. This example provides a Network Controller





BURROUGHS CORPORATION  
COMPUTER SYSTEMS GROUP  
SANTA BARBARA PLANT

8-9  
COMPANY CONFIDENTIAL  
91800/81700 SYCOM  
P. S. 2219 0458 (A)

Z to Server System's SMCS.  
Z rsr 001

FINI.

-----//-----

Source configuration notes:

The Network Controller source file for a Server System "with local terminals" is configured by appropriately adding the NDL statements identified in the User System example with "%\*" as comments.

## OUTPUT AND ERROR MESSAGES

### GENERAL

There are three basic types of output and error messages produced by the SYCOM system and its associated system software. These messages are listed in their order of presentation and identify the basic reasons for the occurrence of these messages:

1. Master Control Program Messages
2. Network Controller Messages
3. SYCOM Error Messages

### MASTER CONTROL PROGRAM MESSAGES

The Master Control Program displays messages in response to a control instruction or to inform the system operator of events that have or are to take place. All MCP output messages start in position 2 of the console printer/display. Refer to the B1800/B1700 Systems System Software Operational Guide for further information regarding these messages.

### NETWORK CONTROLLER ERROR MESSAGES

Messages displayed by the Network Controller on the system ODT generally describe error conditions encountered during normal system operations. The actual output messages are displayed starting in position 2 and have the following format:

<Network Controller name> =<job number> <message text>

The following error messages that can occur while the Controller is executing are listed in alphabetical order and explained as follows:

#### #RETRIES UP ON LINE <number>

The retry limit is exhausted for the current message without being acknowledged by the receiving system. The "<number>" represents the data communications line number as defined in the Network Controller.

#### MESSAGE RECEIVED FOR INVALID ADDRESS

A message has been received with an address other than "00", "01", "02", "03", or "04". The message is discarded.

**MESSAGE RECEIVED FROM DISABLED STATION <lsn>**

A message has been received for a station that there is no more room left to store the outgoing message. The <lsn> represents the station's Logical Station Number as defined in the Network Controller. The message is discarded.

**SYCOM ERROR MESSAGES**

During command processing various errors may occur; the response error messages and their meanings are shown below. There are five classes of these error messages:

- Command Informational/Error Messages
- System Parameter Error Messages
- System Error Messages
- System Condition Messages
- System Fatal Messages

The following error message groups describe all messages that may be displayed in each of the above identified message classes:

**COMMAND INFORMATIONAL/ERROR MESSAGES**

The following informational and error messages are displayed as the result of improper use of the SYCOM commands. Generally, an message can occur for one of three reasons; they are:

1. A existing condition when the command is entered
2. Invalid command syntax
3. Invalid parameter data

Although the reason for each message is recognizable, there has been no attempt to group the messages into those categories. The following command informational and error messages are however, alphabetically grouped by command to make locating easier.

**UNRECOGNIZED COMMAND "<token>"**

This simply means that the last SYCOM command entered was not a valid command.

**NON-ACCEPTABLE 2<character>2 REPLACED WITH BLANK**

An unacceptable character has been entered in a command string which is invalid to SYCOM. This message informs the user that this has occurred and has been corrected.

**SYCOM CLOSING[<message>]; CANNOT ACCEPT INPUT**

This message is returned in response to any new command entered "after" an ".EOJ" command is entered but prior to the actual shutdown of SYCOM. (Once shutdown has been initiated, commands are no longer accepted.)

**ABORT**

**THERE IS NO "PUT" IN PROGRESS TO ABORT**

An ABORT command was entered with a specified option of of "PUT" and no file is being transmitted at this time.

**THERE IS NO "FETCH" IN PROGRESS TO ABORT**

An ABORT command was entered with a specified option of of "FETCH" and no file is being received at this time.

**UNRECOGNIZED OBJECT OF "ABORT" "<token>"**

The command was entered with an option keyword which was neither "PUT" or "FETCH". Try again...

**BUFFER**

**BUFFER SIZE "<token>" EXCEEDS MAXIMUM ALLOWABLE 1950 RUN TIME TRANSMISSION BUFFER SIZE CHANGED TO 1950**

This simply means that the value entered for the new "buffer size" is unacceptable. Therefore, SYCOM not knowing the intention of the change, has automatically set the BUFFER SIZE to its maximum.

**NEW BUFFER SIZE IS LESS THAN 362 RUN TIME TRANSMISSION BUFFER SIZE CHANGED TO 362**

This means exactly what it reads, a value less than "362" was entered as a "buffer size" and SYCOM automatically set the BUFFER SIZE to it minimum. It should be pointed out, however, that this size will be unacceptable for proper SYCOM operations.

**CL**

**UNRECOGNIZED COMMAND "CL <token>"**

The .CL command was entered either without an option or the option was found to be invalid.

**CONNECT**

**STATION ALREADY CONNECTED**

This only means that a ".CONNECT" command was entered from a terminal (or ODT) that was already "connected" to a Server System; no problem.

**"CONNECT FAILED": VIRTUAL TERMINAL TABLE FULL**

This message displayed indicates that all allocated "virtual terminal" positions are currently in use. The terminal user must try again at a later time.

**NOT DONE SINCE OFFLINE OR LOGGED OFF**

This message suggests that the request to "connect" has failed because: 1) the remote SYCOM is OFFLINE or 2) the user is not properly logged-on to SYCOM.

COPY

**CAN'T SCHEDULE NOW; TRY AGAIN LATER**

The "PUT/FETCH queue" is presently full, therefore, no more "PUT's" or "FETCH's" can be accepted at this time.

**REQUEST SCHEDULED; WILL SEND FILE <file name> LATER**

The PUT/FETCH request was queued to disk and will be processed as soon as in-process and previously queued file transfers have been completed.

**REQUEST SCHEDULED--REMOTE SYCOM OFFLINE**

The PUT/FETCH requested has been queued to disk and will be processed as soon as communications have been re-established between the User and Server systems.

**"<token>" UNRECOGNIZED**

No required keyword text has been entered such as "TO", "ONTO", or "FROM", therefore, what was entered in error is displayed in this message.

**INVALID USERCODE "<token>"**

The "usercode" entered was rejected because it contains non-alphabetic characters. What was entered in error is displayed in the error message.

**USERCODE MAY NOT HAVE MORE THAN 17 CHARACTERS**

The length of the "usercode" entered exceeded a maximum allowable length of 17 characters.

**SOURCE HOST "" NOT DEFINED OR REMOTE HOST "" NOT ONLINE**

An invalid "host system name" has been specified. It is applicable to either the "source" or "remote" name.

**INVALID PASSWORD**

The "password" entered was rejected because it contains non-alphabetic characters. This usually caused only by "leading" invalid characters as opposed to imbedded or trailing invalid characters.

**PASSWORD MAY NOT HAVE MORE THAN 17 CHARACTERS**

The length of the "password" entered exceeded a maximum allowable length of 17 characters.

**INVALID FILE ATTRIBUTE: <token>**

The command syntax keyword "US" or "USER" was expected and not found in the command syntax.

**INVALID FILE ATTRIBUTE DELIMITER: "<token>"**

The "COPY" syntax requires a comma "," between the attribute "<usercode>" and the keyword "PASSWORD".

**FILE NAME MISSING**

The command syntax requires a "file name" specified for a file transfer; it was either not given or recognized.

**KIND DEFINED TWICE**

The "KIND" option keyword has been previously specified in the command string.

**INVALID KIND ATTRIBUTE "<token>"**

The kind option keyword was not "PACK", "PETAPE", "TAPE", "TAPE7", or "TAPE9" which makes what was entered invalid.

**INVALID HOST NAME**

The host system name entered was rejected because it contained non-alphabetic characters. This message could apply to either the "source" or "remote" host name.

**HOST NAME HAS MORE THAN 17 CHARACTERS**

When a host system name is entered with more than 17 characters (maximum allowable). This message could apply to either the "source" or "remote" host name.

**UNRECOGNIZABLE VOLUME ATTRIBUTE "<token>"**

This message is somewhat misleading in that it is only an indication that a keyword of "KIND" or "HOST" was expected and not found.

**MISSING "," BETWEEN VOLUME ATTRIBUTES**

An attribute separator character of "," is required by the command syntax to indicate the end of the attribute "KIND" and its associated option keyword. It is not there and it is wanted.

**VOLUME NAME HAS MORE THAN 17 CHARACTERS**

The "volume name" (or pack-id) entered exceeded the maximum allowable length of 17 characters. This message could relate to either "volume-1" or "volume-2" names.

**INVALID SOURCE HOST NAME "<token>"**

The system identifier specified as a "source host" does not match the "local" system name, the "remote" system name, and is not blanks. The request is not processed.

**FILE <file name> NOT OPENED [1] OUTPUT PACK NOT FOUND**

During SYCOM's file open-on-behalf-of processing, the MCP returns a value for the last communicate issued. This value indicates that the required destination pack is not currently online.

**FILE <file name> NOT OPENED [4] INVALID USERCODE/PASSWORD**

The MCP has informed SYCOM that the open-on-behalf-of request has been rejected because the Usercode/Password combination used was found to be invalid.

**FILE <file name> NOT FOUND [1] INPUT FILE MISSING**

The object file for a PUT file transfer was not found in the cisk directory (disk) or was not online (tape or disk pack).

**FILE <file name> NOT FOUND [2] FILE LOCKED**

The MCP has informed SYCOM that the open-on-behalf-of request has been rejected because the target file is currently "locked".

**FILE <file name> NOT FOUND [3] NO USERCODE FILE**

The MCP has informed SYCOM that the open-on-behalf-of request has been rejected because the (SYSTEM)/USERCODE file does not exist; it is required.

**FILE <file name> NOT FOUND [4] INVALID USERCODE/PASSWORD**

The MCP has informed SYCOM that the open-on-behalf-of request has been rejected because the Usercode/Password combination used was found to be invalid.

**FILE <file name> NOT FOUND [5] VIOLATES USERCODE CONVENTION**

The MCP has informed SYCOM that the open-on-behalf-of request has been rejected because the Usercode/Password combination entered is not privileged; as required.

**FILE <file name> NOT FOUND [6] SECURITY TYPE ERROR**

The MCP has informed SYCOM that the open-on-behalf-of request has been rejected because no Usercode/Password was entered when it is required.

**FILE <file name> NOT FOUND [7] SECURITY USE ERROR**

The MCP has informed SYCOM that the open-on-behalf-of request has been rejected because a non-privileged user is trying to access a privileged file.

**FILE <file name> NOT FOUND [8] EXECUTE ONLY FILE**

The file requested for transfer is flagged by the MCP as a non-accessible file for internal execution only and the request has been rejected for that reason.

**FILE <file name> NOT OPENED [9] WOULD BE DISCARDED**

The MCP would discard the file at the end of transmission (e.g. attempt to create a non usercoded file by a non-privileged user).

**FILE <file name> LOCKED**

The object file for a PUT file transfer was found to be "locked" in the disk directory (cisk).

**FILE <file name> FILE TYPE CANNOT BE TRANSMITTED**

The disk file specified for file transfer is not one of the file types which SYCOM can transfer.

**SOURCE OR DESTINATION FILE NAME MISSING**

This message indicates that a subject file name was missing from the command string which was entered as either a "COPY", "PUT", or "FETCH" from another system.

**SOURCE OR DESTINATION FILE NAME "<token>" INVALID**

The specified file name was found to contain characters other than those which are considered valid for a file identifier; or, a multi-file identifier was indicated ("/") but was not included.

**FILE NAME WAS IDENTIFIED WITH MORE THAN 10 CHARACTERS**

The name of the file to be transmitted was found to contain an identifier in excess of 10 characters. Its the maximum allowable.

**CANNOT LIST THIS FILE - NOT A BACKUP FILE**

A file was received from another system and was flagged for "auto print". This message explains the reason for not doing it and would correspond to the "<file name> RECEIVED" CDT message.

**FILE <file name> ALREADY EXISTS AT DESTINATION - PUT ABORTED**

When SYCOM is setting up to "receive" a disk (or pack) file, it checks to disk directory. If the specified file already exists, this message is displayed and the transfer is aborted.

**FILE FAMILY NOT ALLOWED**

As suspected, in the specified file name, an equal sign ("=") was found which represented either a "pack-id", "multi-file-id". This is illegal; the file named must be specific.

**DEBUG**

**DEBUG ALREADY ON**

The command was entered to turn DEBUG "ON" which was currently on.

**DEBUG WAS ALREADY OFF**

The command was entered to turn DEBUG "OFF" which was currently off.

**ERROR - "ON" OR "OFF" OR "DUMP" EXPECTED**

The option keyword entered with the DEBUG command is not valid. Please try again.

**DISCONNECT**

**STATION ALREADY DISCONNECTED**

This only means that a "DISCONNECT" message was entered from a terminal that was already disconnected; it was ignored.

**EOJ**

**STOP OR WAIT IN PROGRESS**

This means that a ".WAIT" command has been entered and SYCOM is responding to that request; the ".EOJ" command is ignored.

**FETCH (also see COPY)**

**CANNOT FETCH WHEN REMOTE SYCOM OFFLINE**

This message is indication that an inter-system line connection does not presently exist and the "FETCH" request cannot be accommodated.

**SOURCE OR HOST NAMES MAY NOT BE DEFINED IN A FETCH**

Use of the shorthand version (FETCH) of the COPY command implies the source and destination systems, therefore, they cannot be specified.

**FILE <file name> IN USE AT DESTINATION--FETCH ABORTED**

The file in question was received and when attempting to close it (which removes any existing file copy), finds that the file to be replaced is currently in use. The FETCH is aborted and the received file is lost; it must be requested again.

**LIST**

**UNRECOGNIZED OPERAND "<token>"**

This simply means that an incorrect or misspelled option keyword was entered; and rejected.

**NO PARAMETER(S) SPECIFIED FOR "LIST"**

The command was entered without any keyword options.

REMOTE SYCOM IS OFFLINE. LAST COUNT OF FETCHES WAS 0  
This message is self explanatory.

#### OFFLINE

##### STOP OR WAIT IN PROGRESS

This message indicates that SYCOM is in the process of servicing a ".WAIT" command which in effect produces an "offline" condition; the ".OFFLINE" command is ignored.

#### PHONE

##### INVALID PHONE NO

This message is returned when the phone number entered contains alphabetic characters.

##### PHONE NUMBER TOO LONG

This message is returned when the phone number entered contains more than 20 digits (including hyphens).

##### PHONE NUMBER REJECTED BY NC "<phone number>"

The Network Controller has rejected the request by SYCOM to change the existing "phone number" to the one entered; no reason was specified for the rejection.

#### PUT (also see COPY)

##### CANNOT PUT WHEN REMOTE SYCOM OFFLINE

This message means that an inter-system connection does not presently exist, therefore, a "PUT" cannot be done. The request, however, is saved in the "PUTQUEUE" for later transmission.

##### SOURCE OR DESTINATION HOST NAMES MAY NOT BE DEFINED IN PUT

Use of the shorthand version (PUT) of the COPY command implies the source and destination systems, therefore, they cannot be specified in the command string.

##### PUT OF FILE <file name> ABORTED

This message occurs when SYCOM is attempting to "send" a file to the other system and the specified file is currently in use on that system. The PUT is aborted.

#### QS

ERROR: ".QS" OPERAND NOT NUMERIC

The ".QS" command is entered with a non-numeric queue size integer field and was ignored. It is necessary to re-enter the command string in correct format.

**ERROR: "QS" OPERAND MAY NOT EXCEED 15**

The queue size integer entered was greater than 15 and was summarily rejected. It must be re-entered with an appropriate integer value.

**ERROR: "QS" OPERAND MUST BE AT LEAST 4**

The queue size integer entered was less than the value 4 and was rejected. The command must be re-entered with a legitimate integer value.

**READ**

**REQUEST IGNORED: CARD FILE STILL OPEN**

This message is displayed if a ".READ" command is entered when an input card file is currently active. The entire command string is rejected and discarded.

**REQUEST IGNORED: NOT LOGGED ON**

This message could mean the card reader is not ready.

**FILE "<file-name>" NOT PRESENT, TRY AGAIN**

The command ".READ" was entered and the card or disk file "<file-name>" is not present.

**RESTART**

**11 consecutive timeouts. Remote SYCOM presumed "OFFLINE" AT 08:22:29.1 ON 16 AUG 79**

This is a direct indication that the remote system is not responding to the local SYCOM procedures and is considered to be in an OFFLINE state.

**RY**

**UNRECOGNIZED COMMAND "RY <token>"**

The command string contained an unrecognizable option keyword; not "CP", "LP", or "CR".

**SIZE**

**ERROR: ".SIZE" OPERAND NOT NUMERIC**

The integer entered with the command string was comprised of at least one non-digit character. The integer must be all digits, no exceptions.

**ERROR: ".SIZE" CANNOT EXCEED 400**

This message is displayed when a size integer is entered which exceeds the maximum message block size established for the system; four hundred (400) bytes.

**ERROR: ".SIZE" CANNOT BE LESS THAN 78**

This message is displayed when the size integer entered is less than the minimum allowable message block size; for this system the minimum size is 78 bytes.

**STOP**

This command cannot directly cause an error message.

**SV**

**UNRECOGNIZED COMMAND "SV <token>"**

The .SV command was entered either without an option or the option entered was found to be invalid (not "LP" or "CP").

**TRACE**

**DISK TRACE ALREADY ON**

The command was entered to turn tracing "ON" and SYCOM was already in trace mode.

**DISK TRACE WAS ALREADY OFF**

The command was entered to turn tracing "OFF" and SYCOM was not currently in trace mode.

**ERROR - "ON" OR "OFF" EXPECTED**

The only valid options to the TRACE command are "ON", "OFF" or no option specified.

**USER**

**INVALID USERCODE "<token>"**

The usercode used to log-on with was found to contain digits. Please log-off and log-on again with correct usercode/password combination.

**INVALID PASSWORD "<token>"**

The password which the user signed-on with was found to contain digits. Please log-off and log-on again with a legitimate usercode/password combination.

**CANNOT SIGN ON IN 8.0 INDIRECT MODE. PROTOCOL 2 WILL USE USERCODE/PASSWORD FROM SMCS IF SIGNED ON**

The user attempted to PASS ".us usercode/password" to SYCOM without being "logged-on" to the SMCS.

## **SYSTEM PARAMETER ERROR MESSAGES**

The following error messages could occur during the parameters input process. The problem indicated applies to both ODT input as well as card input. In either case, SYCOM must be DS'd, the problem(s) corrected, and the parameters reentered as required.

### **INVALID PARAMETER: "<token>"**

This message is displayed when the parameter keyword entered did not match one of the acceptable keywords.

### **"=" EXPECTED BUT GOT "<token>"**

A parameter was entered with a value which must be separated by the equal (=) character.

### **INTEGER EXPECTED BUT GOT "<token>"**

The BUFFER and RECORD parameters require an integer which specifies the BUFFER/RECORD size in bytes. As this is not what was entered, what was found is included in the message.

### **BUFFER SIZE "<token>" EXCEEDS MAXIMUM ALLOWABLE 1950**

The maximum allowable BUFFER size is 1950 bytes. What was specified exceeded this limit.

### **INVALID SYSTEM ID "<token>" TOO LONG**

The "system id" entered has either exceeded the parameter size limit of 17 bytes, it was excluded altogether, or special characters have been detected.

### **INVALID PARAMETERS FILE, BEING REMOVED**

This message occurs when the "PARAMETERS" file is opened, the parameters read in, and the version is incorrect or the "maximum buffer size" was found to exceed the system maximum of 1950 bytes. When this condition is encountered, the "SYCOM/PARAMETERS" file is removed and new parameters must be entered.

## **SYSTEM WARNING MESSAGES**

Warning messages displayed on the User ODT indicate a problem at the system level. They have been rectified by SYCOM so that operations can continue without operator intervention. They are not generally considered to be major or extensive in nature and therefore, SYCOM has elected to take immediate corrective action.

### **### WARNING - REMOTE HOST NAME SAME AS LOCAL SYSTEM ID.**

This problem is resolved by stopping SYCOM and re-creating the "SYCOM/PARAMETERS" file with an appropriate "SYSTEMID" change or using .SYSTEMID command.

**INQ OVERFLOW, FOLLOWING MSG DISCARDED <60 or less characters**

**of message>**

If during a Write to a Program-to-SYCOM application program an "incomplete-i/o" occurs, this message is displayed to provide the warning for this condition.

**TERMINALS FILE OVERFLOW, MESSAGE DISCARDED <60 or less characters of message>**

If during a message Write to the remote file TERMINALS an "incomplete-i/o" occurs, the above message is displayed to inform the system operator of this condition.

**INCOMPLETE IO ON LINK FILE IGNORED**

OR

**INCOMPLETE IO ON TERMINALS FILE IGNORED**

OR

**INCOMPLETE IO ON OUTQ FILE IGNORED**

The complex-wait has interrupted SYCOM for either the LINK, TERMINALS, or OUTQ file and the Result Descriptor indicates an "incomplete i/o" has occurred. This error is logged and SYCOM reverts back to "wait" mode.

**LINK QUEUE OVERFLOW, MESSAGE DISCARDED**

The queue "LINK.Q" is full, therefore, no more messages can be queued at this time. The message in question is dumped as there is nothing SYCOM can do with it.

**NUMBER OF TIMEOUTS = <number>**

This message is displayed automatically when "timeouts" keep re-occurring on the inter-system line connection. It is only output on an incremental basis (n) to reduce ODT activity; where <n> is specified in the "TIMEOUTS" parameter.

**LOSS OF DATA-SET-READY**

Display of this message occurs each time the indicated event is detected. A count is maintained of these occurrences and SYCOM goes into the state of "initialization". In progress file transfers are aborted and lost. If this continuously occurs at BOJ, the adapter may be incorrectly strapped.

**HOST SYSTEM CARD READER HALTED**

When this message occurs it is to inform the system operator of a condition that presently exists on the remote system.

**HOST SYSTEM CARD READER ABORTED**

This message is used to inform the system operator that card reader input from the remote system has been aborted. No action is required of the operator.

**HOST SYSTEM CARD READER READY**

A message that is used to inform the local system operator that card input from the remote system has resumed.

**SYCOM WILL CLOSE WHEN NEXT IDLE**

The "EOJ" command has been entered and SYCOM is processing previously entered requests. When these requests have been processed, SYCOM will terminate.

**## SPURIOUS .CONNECT IGNORED ##**

This meaning is clear; SYCOM was interrupted with an unknown "connect" message which it ignored.

**## SPURIOUS .DISCONNECT IGNORED ##**

The meaning of this message is clear; SYCOM was interrupted with an unknown "disconnect" message which it ignored.

**SYCOM CLOSING ON REQUEST FROM REMOTE SYCOM**

The remote SYCOM program has been directed to go to "EOJ" by the entry of either "EOJ" or "STOP" from that system. When this occurs, it sends a control type message to the other SYCOM which causes the above message to be displayed.

**ERROR: MESSAGE TYPE = <type code>**

An input message from the remote file "LINK" was found to be invalid. The message type code is included in the displayed error message for reference.

**REMOTE SYCOM HAS GONE DOWN AND COME BACK UP**

This message is self explanatory in that for some reason the remote SYCOM disconnected the line and then re-established communication with this SYCOM resulting in the above message being displayed on the system ODT.

**QUEUE RJEQ FULL, MESSAGE DISCARDED**

The queue "RJE.Q" is full, therefore, no more messages can be queued to it and the current message is discarded.

**### WARNING - NO "SYSTEMID" ###**

Upon completion of the accept SYCOM parameters process, it was determined that a "system-id" has not been specified (either not entered or was in error).

**SYSTEM CONDITION MESSAGES**

Any of the following messages displayed on the User ODT indicate an abnormal system condition of some type. As a rule, when one of these conditions is encountered, the display of an appropriate message also elicits a response from the system operator.

**CARD PUNCH EXCEPTION PLEASE ENTER ".RY CP" WHEN DEVICE IS READY OR ".SV CP" TO SAVE THE FILE**

This message is displayed when SYCOM is attempting to punch a card and an "exception" condition is encountered. Generally this indicates that the card punch has gone "not ready" for any one of several reasons. It is necessary to respond to SYCOM with the appropriate command when action has been taken to ready the unit.

**LINE PRINTER EXCEPTION PLEASE ENTER ".RY LP" WHEN DEVICE IS READY OR ".SV LP" TO SAVE THE FILE**

This message is displayed when SYCOM is attempting to print a line and an "exception" condition is encountered. Generally this indicates that the line printer has gone "not ready" for any one of several reasons. It is necessary to respond to SYCOM with the appropriate command when action has been taken to ready the unit.

**PARITY ERROR ON READ PLEASE READ LAST CARD AGAIN**

This message is displayed when SYCOM is attempting to read the card file and an "exception" condition is encountered. Generally this indicates that the card reader has gone "not ready" for any one of several reasons. It is necessary to respond to SYCOM with the appropriate command when action has been taken to ready the unit.

**<message text>: BAD LSN = <message lsn>**

When writing a reply message to a terminal, an "EOF" status was encountered and SYCOM assumed that an invalid "lsn" was the cause. Therefore, that message text is displayed on the System ODT as part of the above message. No further action is taken on the part of SYCOM.

**<message text>: INVALID LSN = <message lsn>**

When writing a reply message to a terminal, an "EOF" status was encountered and SYCOM assumed that an invalid "lsn" was the cause. Therefore, that message text is displayed on the System ODT as part of the above message. No further action has been taken by SYCOM in this matter.

**PUNCH ERROR**

The card punch has either detected an error in the last card punched or has degraded into a "not ready" state. In either case, a "suspend" control message has been sent to the Server System.

#### READER ERROR

The card reader has either detected an error in the last card read, gone into a "not ready" state, or has detected an invalid character in a card column; other than column 1. If there was a read error, the last card must be put back into the card hopper. The card reader is then made physically "ready"; all cases. SYCOM must then be informed of this new status condition with the entry of a ".READY" command.

#### PRINTER ERROR

The line printer has either detected an error on the last line printed or has gone into a "not ready" state. In both cases a "suspend" control message is sent to the Server System. When everything is again ready with the printer, a ".READY" command must be entered to inform SYCOM.

#### CARD READER CLOSED

This simply means the card input has been completed and the card file closed.

#### QC INITIATED

In doing a read on the remote file "LINK" an "EOF" condition was detected. This condition is caused by the MCP when a "QC" command is entered; which shuts down all programs associated with the Network Controller.

#### NO PHONE NUMBER

This is indication that when attempting to establish communications with another system, SYCOM determined "auto dial-up" was necessary but the required phone number is not present.

#### INVALID CONTROL MESSAGE RECEIVED = <message text>

When SYCOM receives a system "control message" from a remote system and the "type code" in this message is not within the 00 - 23 number range, then the above message is displayed. The input message is discarded and no further processing is done on its behalf.

#### BUG IN NETWORK CONTROLLER. VTT INDEX = <vtt> I.E. STATION WAS NOT CONNECTED OR WRONG LSN IN CLOSE MESSAGE.

A "close" message has been received from the system Network Controller for a program known to SYCOM which contained a "lsn list". This list included a Virtual Terminal Table Index (VTT INDEX) entry for a terminal not signed-on to that program, thus the reason for this message.

#### REMOTE SYCOM TERMINATED

This message indicates that the communications link between the two systems has been interrupted for some reason. SYCOM will attempt to re-establish the connection and failing that reverts to "Offline Standby" mode.

SYSTEM FATAL MESSAGES

These messages are of the type which preclude the automatic shut down of SYCOM. There are considered "fatal" in the outlook of continuing operations and therefore, SYCOM is shut down.

**FATAL ERROR AT <program sequence number> <message>**

The display of this message on the User ODT could be the result of any number of reasons. The SYCOM source sequence number displayed is the only clue to the problem. SYCOM initiates a "list-disk-trace" and a "memory dump" at this and automatically goes to EOJ after certain disconnect functions have been satisfied.

**REMOTE VERSION OF SYCOM IS INCOMPATIBLE HIS VERSION="<version number>", MINE="02"**

This message identifies the problem at hand. SYCOM goes to EOJ immediately at this point. This version is determined by the SDS.

**ANOTHER SYCOM WITH SAME MULTI-FILE-ID IS RUNNING**

The display of this message indicates that SYCOM cannot gain access to the "SYCOM/PUT.QUELE" file (on disk) as it is presently in use by another program.

**REMOVE PUT.QUEUE OR CHANGE ITS NAME IF 7.0 FILE**

If the disk file by that name is less than 99 records, when reading it, SYCOM will get an "EOF" and aborts immediately.

APPENDIX B: CHARACTER SET

The following graphic character set defines those characters that are acceptable to SYCOM as input. These characters can be in the form of terminal input/output messages, data files or intersystem control messages. Other characters encountered by SYCOM will be dealt with as per the established "file transfer" rules.

The graphic character set is arranged in "rows" by the internal collating sequence with "blank" being lowest and "9" high:

|       |   |   |   |   |   |   |   |   |    |   |   |   |
|-------|---|---|---|---|---|---|---|---|----|---|---|---|
| blank | [ | . | < | ( | + |   | & | ] | \$ | * | ) | ; |
| -     | - | / | π | , | z | _ | > | ? | :  | # | a | ' |
| =     | " | a | b | c | d | e | f | g | h  | i | j | k |
| l     | m | n | o | p | q | r | s | t | u  | v | w | x |
| y     | z | A | B | C | D | E | F | G | H  | I | J | K |
| L     | M | N | O | P | Q | R | \ | S | T  | U | V | W |
| X     | Y | Z | 0 | 1 | 2 | 3 | 4 | 5 | 6  | 7 | 8 | 9 |

It is recommended that the standard "signal character" of "." be retained by the SYCOM for use as a local SYCOM message switching character. However, if this is not suitable for any reason, one the following special characters is suggested: (A very important point to note is that when changing SYCOM's "signal character", do assign a character which conflicts with the system's current "SMCS signal character".)

[ & \$ % ] # a +

Special data communications characters recognized by SYCOM are:

SOH EOT DLE ENQ STX ETX RS FS

APPENDIX C: COMMAND SUMMARY

The following command summary list is presented in alphabetical sequence for easy reference. Refer to the command descriptions for command syntax and option details.

| COMMAND    | STATION | FUNCTION                                                                    |
|------------|---------|-----------------------------------------------------------------------------|
| ABORT      | any     | Abort a PUT or FETCH file transfer.                                         |
| ADD        | any     | Transfer a file if it does not already exist at the destination             |
| BUFFER     | any c   | Query or change transmission block size.                                    |
| BYE        | any     | Station log-off from SYCOM.                                                 |
| ICL        | ODT     | Clear peripheral device.                                                    |
| CONNECT    | any     | Request establishment of a terminal to a Server System.                     |
| COPY       | any     | Transfer a file (regardless of whether or not it exists at the destination) |
| ICP        | ODT     | Inform SYCOM of "not ready" peripheral.                                     |
| DEBUG      | any q   | Start/stop input/output message tracing.                                    |
| DISCONNECT | any     | Request for terminal disconnection from the Server System.                  |
| EOJ        | ODT     | Cause SYCOM into an orderly shutdown.                                       |
| FETCH      | any     | File transfers (fetch from other system)                                    |
| GROUPSIZE  | any q   | Set or interrogate group size for transfers of families of files            |
| LIST       | any     | Display selective system information.                                       |
| LOG        | any q   | Display a list of data comm error counts and program statistics             |
| ILP        | ODT     | Inform SYCOM line printer "not ready".                                      |
| OFFLINE    | ODT     | Cause SYCOM to break the connection to a Server System.                     |

|            |  |       |  |                                          |  |
|------------|--|-------|--|------------------------------------------|--|
| I ONLINE   |  | ODT   |  | Establish a Server System connection.    |  |
|            |  |       |  |                                          |  |
| I PHONE    |  | any q |  | Establish or clear phone number.         |  |
|            |  |       |  | Server System.                           |  |
|            |  |       |  |                                          |  |
| I PROGRAM  |  | any q |  | queue or change program parameter        |  |
|            |  |       |  |                                          |  |
| I PUT      |  | any   |  | File transfers (put to other system).    |  |
|            |  |       |  |                                          |  |
| I QS       |  | ODT q |  | Query or change Network Controller queue |  |
|            |  |       |  | depth size.                              |  |
|            |  |       |  |                                          |  |
| I READ     |  | ODT   |  | Open and transmit card input file.       |  |
|            |  |       |  |                                          |  |
| I RESTART  |  | ODT   |  | Disconnect line, initialize and standby. |  |
|            |  |       |  |                                          |  |
| I RY       |  | ODT   |  | Inform SYCOM of available peripheral.    |  |
|            |  |       |  |                                          |  |
| I SIGNAL   |  | any   |  | Query or change SYCOM station's pseudo   |  |
|            |  |       |  | signal-character.                        |  |
|            |  |       |  |                                          |  |
| I SIZE     |  | any q |  | Change maximum message block size.       |  |
|            |  |       |  |                                          |  |
| I STOP     |  | ODT   |  | Cause SYCOM to shutdown immediately.     |  |
|            |  |       |  |                                          |  |
| I SV       |  | ODT   |  | Stop transmitting and "save" file.       |  |
|            |  |       |  |                                          |  |
| I SYSTEMID |  | any q |  | Interrogate or change value of SYSTEMID  |  |
|            |  |       |  | parameter                                |  |
|            |  |       |  |                                          |  |
| I TRACE    |  | any c |  | Start/Stop SYCOM I/O disk tracing.       |  |
|            |  |       |  |                                          |  |
| I USER     |  | any   |  | Log a user onto SYCOM.                   |  |
|            |  |       |  |                                          |  |
| I WAIT     |  | any q |  | Interrogate or change value of WAIT      |  |
|            |  |       |  | parameter                                |  |
|            |  |       |  |                                          |  |
| I WHAT     |  | any   |  | SYCOM status inquiry.                    |  |

---

### APPENDIX D: SYNTACTICAL KEYWORD SUMMARY

Syntactical keywords will be recognized by SYCOM when they appear within specific context within the command string syntax. Should a specified "file name" happen to match one of these "keywords", do not worry, for this is a perfectly acceptable use of a keyword outside of the command syntactical context. A complete summary of these keywords follows:

|            |           |          |       |          |
|------------|-----------|----------|-------|----------|
| AS         | BINARY    | CLEAR    | CP    | CR       |
| DISK       | DUMP      | FETCH    | FROM  | HOST     |
| KIND       | LIST      | LP       | NO    | NR       |
| OFF        | OLDTRACE  | ON       | ONTO  | PACK     |
| PARAMETERS | PASSWORD  | PETAPE   | PUT   | PUTQUEUE |
| REMOVE     | SAVE      | SCHEDULE | TAPE  | TAPE7    |
| TAPE9      | TERMINALS | TO       | TRACE | US       |
| USER       | USERS     |          |       |          |

## APPENDIX E: COMMUNICATIONS CONTROL MESSAGES

### GENERAL

As there are several message formats used for inter-SYCOM data communications, each format is specified in detail. These message formats are presented in a sequence dictated by the message type code. A list of message types are shown below, followed by individual message format descriptions.

### MESSAGE TYPES

The message type codes are derivatives from the "DA1"/"DA2" (Device Address) field. The message type codes apply to message transmissions in either direction and are compatible between system types. Following then are the device address conventions used by SYCOM; all other values are considered invalid:

| DA1      | DA2      | Usage                                                                      |
|----------|----------|----------------------------------------------------------------------------|
| 0        | 0        | System control                                                             |
| 0        | 1        | ODT messages                                                               |
| 0        | 2        | Card reader/punch data messages                                            |
| 0        | 3        | Line printer data messages                                                 |
| 0        | 4        | File transfer data messages                                                |
| 8        | 2        | Packed card punch data messages                                            |
| a thru z | A thru Z | User System "transmit", Server System "receive", virtual terminal messages |
| A thru Z | a thru z | User System "receive", Server System "transmit", virtual terminal messages |

Device addresses are advisory only. The receiving system can direct a message to any appropriate peripheral device, based on its own requirements.

Note: Line printer data is allowed in both directions. When a system receives line printer data, it can print it on any available printer, independently of whether the system is termed a User System or a Server System.

## MESSAGE FORMATS

### BASIC DATA COMMUNICATION MESSAGES

As per the line protocol, the below message format adheres to the criteria of standard RJE message formatting:

```
SATSDD EB
ORMTAA<----- TEXT ----->TC
HM#X12 XC
```

The individual characters in this message function as follows:

- A. SOH - This is the standard ASCII character which is used to indicate data transmission as opposed to connection establishment communication.
- B. ARM - The "Acknowledge Received Message" character which is used to accept or reject the last message block received.
- C. TM# - The "Transmission Number" character used in the acknowledgement of a message. It represents one of two values (alternating characters) selected for implementation by SYCOM. The characters "5" and "&" are used by the User System SYCOM whereas the characters "F" and "U" are used by the Server System SYCOM.
- D. STX - A standard control character used to delimit the preceding header characters from the message.
- E. DA1/DA2 - Two characters representing a "Device Address" value.
- F. TEXT - The message text field usually containing data (record images or a message).
- G. ETX - The standard "end-of-text" control character.
- H. BCC - A Block Check Character which is comprised of a longitudinal parity bit configuration that is used by the hardware to verify that the message has been received correctly.

### SYSTEM CONTROL MESSAGE

System control messages are used to exchange control type data between systems. A complete descriptive list follows the message format which is a variation of the basic message format:

```
SATSDDCC EB
ORMTAADD<---- variables (text) ---->TC
HM#X1212 XC
```

"00"

Here we have the addition of the "CD1"/"CD2" characters. These two characters represent the system message "type" code; which may or may not be followed by variable control data.

#### ODT MESSAGES

The ODT Message format uses the basic format without alteration. There are no provisions for blocking ODT Messages, therefore, the rule is: one ODT Message per control message block. The format of this type message is as follows:

```
SATSDD EB
ORMTAA<----- TEXT ----->TC
HM#X12 XC
```

"01"

#### CARD READER/PUNCH MESSAGE

The format for card reader/punch data images is as follows: The "RS" (Record Separator) character used between each card image provides the means for blocking.

```
SATSDD EB
ORMTAA<---- card image ---->R<---- card image ---->TC
HM#X12 XC
```

"02"

#### LINE PRINTER MESSAGE

The following format variation is used to transmit Line Printer data lines between systems:

```
SATSDDP EB
ORMTAAC<---- line image ---->R<---- line image ---->TC
HM#X12C XC
```

"03x"

This format is the same as the Card Reader/Punch format except that the "PCC" character is utilized. The value of this field can be one of the following codes:

| PCC | Function                        |
|-----|---------------------------------|
| 0   | Print, no spacing               |
| 1   | Print, single space             |
| 2   | Print, double space             |
| A   | Skip to channel 1 (top of page) |
| B   | " 2                             |
| C   | " 3                             |
| D   | " 4                             |
| E   | " 5                             |
| F   | " 6                             |
| G   | " 7                             |
| H   | " 8                             |
| I   | " 9                             |
| J   | " 10                            |
| K   | " 11                            |
| L   | " 12 (bottom of page)           |
| H   | " 8                             |

#### FILE TRANSFER MESSAGE

The File Transfer message format is also the same as that used to transfer card images. Although the "PCC" character field is included, it is only there to maintain message conformity:

```
SATSDD F EB
DRMTAA<---- record data ---->S<----record data ---->TC
HM#X12 XC
```

"04"

#### VIRTUAL TERMINAL MESSAGE

The message format utilized for virtual terminal messages is the basic message format, with one exception: the "device address" for a virtual terminal is comprised of two letters (1 lower, 1 upper "transmit": 1 upper, 1 lower "receive") and will vary by terminal. The format is as follows: (Terminal and ODT messages are "not" blocked.)

```
SATSDD EB
ORMTAA<----- message text ----->TC
HM#X12 XC
```

"xX" (transmit)  
"Xx" (receive)

#### FILE SEPARATOR MESSAGES

This message format is utilized by SYCOM when it is necessary to transfer a "symbolic file header", indicate the "beginning of a file", or indicate the "end of a file". This applies equally to the "Card Reader/Punch", "Line Printer", and "File Transfer" operations. To accomplish this, there are three minor variations to the File Transfer Message format which are identified with the following message formats.

The "Card Reader/Punch", "Line Printer", and "File Transfer"

#### Begin File Message

The format for the Begin File message is as follows:

```
SATSDDF EB
ORMTAAS1<---- file name ---->TC
HM#X12 XC
```

"02"  
"03"  
"04"

#### End File Message

The format for the "end file" message is as follows:

```
SATSDDF EB
ORMTAAS2<--- File-Name --->TC
HM#X12 XC
```

PROGRAM-to-PROGRAM MESSAGES

The following message format pertains to SYCOM's "program-to-Program" facility; thru a "Remote File":

REMOTE.KEY (10 characters) % Message "header".

- RSN ( 4) % Relative Station Number for  
% "virtual terminal" as defined  
% for the program-to-program  
% remote file; usually "0001".
- LENGTH ( 4) % Length of message text.
- TYPE ( 2) % Message type; usually "00".

MESSAGE TEXT (nn characters) % Maximum is "buffersize".

- LENGTH ( 4) % Length of message text.

PROGRAM-to-SYCOM MESSAGES

SYCOM has the inherent capability to accept command strings from an application program and respond with reply messages thru the "Program-to-SYCOM" interface facility. This facility utilizes the "cueue" mechanism, whereby SYCOM reads input from the queue file "SYCOM/OUTQ" and writes replies into the queue file "SYCOM/INQ". (If these queue file names seem reversed, its because they are oriented to the application program.) Input/Output message formats are defined as follows:

Input (SYCOM/INQ):

|                      |      |                 |
|----------------------|------|-----------------|
| Program supplied I/D | 4    | bytes           |
| Filler               | 6    | bytes           |
| SYCOM command string | 1940 | bytes (maximum) |
| -----                |      |                 |
|                      | 1950 | bytes (maximum) |

Output (SYCOM/OUTQ):

|                      |      |                 |
|----------------------|------|-----------------|
| Program supplied I/D | 4    | bytes           |
| Reply message length | 4    | bytes           |
| Reply message code   | 2    | bytes           |
| SYCOM reply message  | 1940 | bytes (maximum) |
| -----                |      |                 |
|                      | 1950 | bytes (maximum) |
| Reply message length | 4    | bytes           |

## APPENDIX F: RESTRICTED MCP COMMAND SUMMARY

All input commands that are prefixed with a "?" are sent directly to the MCP exactly as entered. The MCP then generates appropriate responses which are sent to the terminal via the Network Controller. Security restrictions have imposed certain constraints on MCP commands from remote terminals. Also, certain commands, such as those which alter system options, purge, or clear system peripherals, have been disallowed for a remote terminal. The following list represents the MCP command subset permitted from User System terminals and the User ODT.

The following Command Subset is listed in the order as described in the B1800/B1700 Systems System Software Operational Guide (associated documentation). Any command listed below which is suffixed with "US" requires the command to be entered with a valid "usercode"/"password" combination.

### Library Maintenance Instructions

- . CH Change (US)
- . AD Add (US)
- . LO Load (US)
- . DU Dump (US)
- . UN Unload (US)
- . RE Remove (US)

### Program Control Instructions

- . CO Compile (US)
- . DY Dynamic
- . EX Execute (US)
- . MH Modify Header (US)
- . MO Modify (US)
- . CO Compile (US)

### Program Control Instruction Attributes

- . AF After
- . AN After.Number
- . CA Conditional
- . CG Charge (certain restrictions apply)
- . DS Dynamic.Spaces
- . FI File
- . FR Freeze
- . HO Hold
- . IN Interp interpreter
- . IT Intrinsic.Name
- . ID Intrinsic.Directory
- . ME Memory

- . OV Override
- . PR Priority
- . SC Schedule-Priority
- . SW Switch
- . TH Then
- . UC Unconditional
- . UF Unfreeze
- . UV Unoverride
- . VI Virtual.Disk

#### System Control Instructions

- . AX Response to ACCEPT
- . BF Display BACKUP files (US)
- . CP Compute
- . CU Core usage
- . DF Date of file (US)
- . DM Dump memory and continue
- . DP Dump memory and discontinue
- . DS Discontinue program
- . FN Display internal file name
- . FR Final reel of unlabeled tape file
- . FS Force from schedule
- . GD Resume stopped program
- . HS Hold in waiting schedule
- . HW Hold in waiting schedule until job EOJ
- . MR Close output file with purge
- . MX Display MIX
- . OF Optional file response
- . OK Continue processing
- . OL Display peripheral status
- . PD Display directory
- . PM Print memory dump
- . PR Change priority
- . OK Continue processing

#### System Control Instructions (continued)

- . PS Prod schedule
- . QF Query file (US)
- . QP Query program
- . RB Remove backup files (US)
- . RM Remove duplicate disk file
- . RS Remove jobs from schedule
- . SP Change schedule priority
- . ST Suspend processing
- . SW Set switch
- . TD Time and date
- . TI Time interrogation
- . TS Test switches
- . UL Assign unlabeled file
- . WD Display MCP date
- . WM Display current MCP and interpreter

- WS Display schedule
- WT Display MCP time
- WY Program status interrogation

Job Spawning Control Attributes

- QU Queue command
- US Usercode command
- ZQ Zip cueue command

MIX related MCP commands may refer orly to jobs initiated by the same remote terminal where the initial command was input. All MCP commands must be entered in UPPER CASE.

APPENDIX G: PARAMETERS FILE ATTRIBUTES

The PARAMETERS file is a disk file, data type, with one physical record of 180 characters. This parameters record is formatted as shown below.

RECORD FORMAT

| Field            | Length<br>(in bits) | Default<br>Value | Description                                                                                        |
|------------------|---------------------|------------------|----------------------------------------------------------------------------------------------------|
| Version          | 8                   | "e"              | Current version of file.                                                                           |
| Filler           | 96                  | " "              | Reserved for identifier.                                                                           |
| CURRENT.DATE     | 72                  |                  | Run date as "dd mm yy"; (9 bytes).                                                                 |
| STATION.ID       | 136                 | " "              | System identifier; 17 bytes.                                                                       |
| MAX.BUFFER.SIZE  | 24                  | 1950             | File Transfer buffer size.                                                                         |
| MAX.MESSAGES     | 24                  | 4                | NDL queue size.                                                                                    |
| TIMEOUT.INTERVAL | 24                  | 0                | A value for message display intervals.                                                             |
| DEBUGGING        | 1                   | 0                | DEBUG on/off switch. Default of "0" inhibits debugging.                                            |
| DISK.TRACING     | 1                   | 1                | Controls disk tracing; 0=no 1=yes.                                                                 |
| NOFS             | 1                   | 0                | When "set", this parameter inhibits File Separators.                                               |
| NOFTP.P          | 1                   | 0                | When "set", this parameter permits File Transfers to occur; else they are denied.                  |
| QUE.P            | 1                   | 0                | When "set", this parameter permits "program-to-SYCOM" communications. occur; else they are denied. |

|                                | Length<br>(in bits) | Default<br>Value | Description                                                                                                   |
|--------------------------------|---------------------|------------------|---------------------------------------------------------------------------------------------------------------|
| WAIT.P                         | 1                   | 0                | To shutdown or not shutdown when receiving a "disconnect message" from another SYCOM. 0 = shutdown; 1 = idle. |
| Filler                         | 2                   | 00               | Reserved for future use.                                                                                      |
| Default N input buffers        | 8                   |                  |                                                                                                               |
| TRACE.R.B                      | 16                  | 4                | Disk trace; records per blk.                                                                                  |
| Default N output buffers       | 8                   |                  |                                                                                                               |
| TRACE.B.A                      | 16                  | 10               | Disk trace, blocks per area. PSEUDOsignal character                                                           |
| Default Pseudosignal character | 8                   | "."              |                                                                                                               |
| TRACE.AREAS                    | 8                   | 10               | Disk trace, number of areas.                                                                                  |
| Filler                         | 8                   | 00               | Reserved for future use                                                                                       |
| GROUPSIZE                      | 24                  | 9                | Size of groups within a family of the transfers                                                               |
| TERMINALS.RF#                  | 24                  | 000              | Remote File Number for the file TERMINALS; 3 bytes.                                                           |
| SYCOM prog name                | 240                 |                  | SYCOM's name; 30 bytes.                                                                                       |
| SYCOM job number               | 32                  |                  | SYCOM'S "job #"; 4 bytes.                                                                                     |
| Filler                         | 256                 |                  | Reserved for identifier.                                                                                      |
| COMPILE.DATE                   | 72                  | " "              | Compile date as "dd mm yy"; (9 bytes).                                                                        |
| Filler                         | 256                 | " "              | Reserved for identifier.                                                                                      |
| COMPILE.TIME                   | 80                  | " "              | Compile time as "hh:mm:ss.t" (10 bytes).                                                                      |
| Filler                         | 88                  |                  | Reserved for future use....                                                                                   |
| Local Release I/D              | 72                  |                  | Release version identifier of the local SYCOM program.                                                        |
| Remote Release I/D             | 72                  |                  | Release version identifier of the remote SYCOM program.                                                       |
| Filler                         | 16                  |                  | Available.                                                                                                    |

APPENDIX H: SUMMARY OF FILES

| File Number | Internal Name | External Name      | Device    | Records/Block |
|-------------|---------------|--------------------|-----------|---------------|
| 0           | DEBUG         | <sycom>/DEBUG      | printer + | 132/1         |
| 1           | DISK_TRACE    | <sycom>/TRACE      | disk      | 132/4         |
| 2           | FILE_NAMES    | FILE_NAMES         | disk      | 72/1          |
| 3           | FILENAME_Q    | FILENAMES          | queue     | 180/1         |
| 4           | IN            | IN *               | disk/tape | *180/1*       |
| 5           | IN_OUT        | <sycom>/IN_OUT     | PBD       | 132/1         |
| 6           | IN_Q          | <sycom>/INQ        | queue     | 2000/1        |
| 7           | LINK          | LINK #             | remote    | 2000/1        |
| 8           | LINK_Q        | <sycom>/LINK.Q     | queue     | 2000/1        |
| 9           | LIST          | <sycom>/LIST.TRACE | printer   | 132/1         |
| 10          | NEWPARAMS     | <sycom>/NEWPARAMS  | card/disk | 80/1          |
| 11          | OLDTRACE      | <sycom>/OLDTRACE   | disk      | 132/4         |
| 12          | OUT           | OUT **             | disk/tape | *180/1*       |
| 13          | OUT_Q         | <sycom>/OUTQ       | queue     | 2000/1        |
| 14          | PARAMETERS    | <sycom>/PARAMETERS | disk      | 180/1         |
| 15          | PUT_QFILE     | <sycom>/PLT.QUEUE  | disk      | 358/1         |
| 16          | RJE_Q         | <sycom>/RJE.Q      | queue     | 455/1         |
| 17          | RJECARDS      | RJE/CARDS #        | card      | 80/1          |
| 18          | RJEPRINT      | RJE/PRINT          | printer + | 132/1         |
| 19          | RJEPUNCH      | RJE/PUNCH          | punch +   | 80/1          |
| 20          | STATISTICS    | <sycom>/STATISTICS | disk      | 180/1         |
| 21          | TERMINALS     | TERMINALS          | remote    | 2000/1        |

\* - These files are changed to the actual device type.

record size, and file name specified in the file  
transfer command.

- + - Backup allowed.
- # - Or user specified name.
- - Use input blocking.

---

\* Note: Printer or printer backup cisk file.  
\*\* Note: The actual file name as specified in the COPY command.

### APPENDIX I: SAMPLE APPLICATION PROGRAMS

This program was designed basically to test "program-to-program" communication functions of SYCOM. What this sample application program provides is the mechanics of what is necessary for this communication process. The user can expand upon this sample to create a program with the desired capabilities as required. The program is written in UPL which is available for this purpose.

The only data communication type functions required for this type of application program is to "read" input messages from a Remote File and "write" reply messages into the same Remote File. This message interfacing architecture has been established for this purpose and must be, therefore, adhered to at all times.

PROGRAM-to-PROGRAM (User System)

COMPILE USER/TEST UPL LI;

& DETAIL  
& SIZE  
& PASS\_END

FILE RFUSER (DEVICE = REMOTE, NUMBER\_OF\_STATIONS = 1,  
REMOTE\_KEY, RECORDS = 80/1, BUFFERS = 2);

DEFINE CH AS #CHARACTER#;

DECLARE TEXT CH (80),  
REPLY CH (80);

DECLARE 1 RMT\_KEY CH (10),  
2 RMT\_RSN CH ( 3),  
2 RMT\_LENGTH CH ( 4),  
2 RMT\_TYPE CH ( 3),

1 READ\_KEY CH (10),  
2 READ\_RSN CH ( 3),  
2 READ\_LENGTH CH ( 4),  
2 READ\_TYPE CH ( 3);

RMT\_RSN := "001"; % First station in "family".  
RMT\_LENGTH := "0080"; % Standard message length.  
RMT\_TYPE := "000"; % MCS type code.

OPEN RFUSER INPUT, OUTPUT;

```
DC TEST_LOOP FOREVER;

ACCEPT TEXT;

WRITE RFUSER [RMT_KEY] (TEXT); % Outputs messages
 % directly to the
 % Network Controller
 % for delivery to the
 % other system.

READ RFUSER [READ_KEY] (REPLY); % Obtains input directly
 % for the Network
 % Controller.

ON EOF STOP; % In case of "QC"

DISPLAY REPLY;

END TEST_LOOP;

FINI;
-----//-----
```

PROGRAM-to-PROGRAM (Server System)

COMPILE SERVER/TEST UPL LI;

& DETAIL  
& SIZE  
& PASS\_END

FILE RFSERVER (DEVICE = REMOTE, NUMBER\_OF\_STATIONS = 1,  
REMOTE\_KEY, RECORDS = 80/1, BUFFERS = 2);

DEFINE CH AS #CHARACTER#;

DECLARE TEXT CH (80),

1 RMT\_KEY CH (10),  
2 RMT\_RSN CH ( 3),  
2 RMT\_LENGTH CH ( 4),  
2 RMT\_TYPE CH ( 3),

1 READ\_KEY CH (10),  
2 READ\_RSN CH ( 3),  
2 READ\_LENGTH CH ( 4),  
2 READ\_TYPE CH ( 3);

RMT\_RSN := "001"; % First station in "family"  
RMT\_LENGTH := "0080"; % Standard message length

```
RMT_TYPE := "000"; % MCS type code

OPEN RFSERVER INPUT, OUTPUT;

DO TEST_LOOP FOREVER;
 READ RFSERVER [READ_KEY] (TEXT); % Input from User AP
 ON EOF STOP; % In case of "QC"..
 WRITE RFSERVER [RMT_KEY] (TEXT); % Output to User AP
END TEST_LOOP;
```

FINI;

-----  
/////

## GLOSSARY OF TERMS AND ACRONYMS

Systems Communications Module (SYCOM) 20 Dec 1979 B1800 Series ■  
Software Operational Guide Page - 210 Section VIII Glossary of  
Terms and Acronyms (Revision "C").

This glossary defines SYCOM terms and acronyms used throughout  
this software operational guide. For ease of use, the glossary  
is separated into two sections; Terms and Acronyms. Each  
section is sequenced alphabetically for quick reference.

### TERMS

#### TERM : DEFINITION

##### **ACTIVE STATION:**

A work station (terminal) that is currently eligible for  
entering or accepting messages.

##### **AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII):**

This code, established by the American Standards  
Association, defines codes for a set of characters to be  
used in the inter-change between business equipment over  
telephone and telegraph circuits. The code consists of 128  
graphic and control characters.

##### **APPLICATION PROGRAM:**

A program that processes data and is unique to one type of  
application. It is sometimes referred to as a "user  
program".

##### **AUTO CALL UNIT:**

A device furnished by the communications common carrier  
which allows a User System to automatically establish a  
dialed link to a Server System over a communications  
network.

TERM : DEFINITION

**BEGINNING OF JOB (BOJ):**

The execution of a single program unit to be performed by the system.

**BINARY CARD:**

A punched card with multiple punches in any one or more columns which represent a non-ASCII character.

**BIT**

The smallest unit of information in a binary system.

**BITS PER SECOND (bps):**

The expression bits-per-second is used to express the speed at which bits of data are transmitted over a communications channel or internally moved within a computer system.

**BLOCK:**

A group of characters/bits transmitted as a single unit of data. Also, see Block Check Character.

**BLOCK CHECK CHARACTER (BCC):**

The Block Check Character is a character appended to data blocks transmitted for the purpose of error detection and parity control.

**BLOCKING FACTOR:**

The maximum number of characters allowed per single transmission.

**BROADCAST:**

The simultaneous dissemination of information to a number of work stations (terminals).

**BUFFER**

This is a temporary storage area that is used during the transfer of data from one device to another. It is also used to compensate for variations in bit rates or timing of events within the system.

**CARRIAGE CONTROL:**

Controls the vertical paper motion for a line printer.

**CHANNEL:**

A device that directs the flow of data between a computer's main memory and input/output devices.

TERM : DEFINITION

**CHARACTER:**

A digit, letter, or special character. (Letters can be both upper or lower case.)

**COMMON CARRIER:**

A company, such as Bell Telephone, that provides communication lines as a public service.

**COMMON CARRIER EXCHANGE:**

A defined geographical area, served by a communications carrier, within which the carrier provides this service.

**COMMUNICATION LINE:**

Any medium, such as a wire or a telephone circuit, that connects a terminal with the computer.

**CONDITIONED LINES:**

These are private and leased transmission lines that are conditioned to reduce data distortion while providing transmission of data at lower error rates.

**CONNECTION:**

An established data transmission path between two or more computer systems. It is considered either a temporary or permanent connection, depending on whether it is established with or without switching facilities, respectively.

**CONTROL CARD:**

A punched card which contains information which identifies a job or describes its requirements to the MCP. The first column must contain an invalid character (80 column cards) or a question mark "?" (96 column cards).

**CONTROL STATE:**

The term "control state", for this system, refers to a program that can assume control of the system's processor with privileged operands. As a rule, the type of control-state program suggested here infers an "operating system" or MCP.

**CONVERSATIONAL MODE:**

An operational mode in which message data (full or partial messages) is used in lieu of, or in addition to, control characters as replies for message information.

**DATA**

Data is any representation of information, such as graphic characters, to which meaning might be assigned.

TERM : DEFINITION

DATA COMMUNICATIONS:

Data communication is a method of transferring encoded data by means of electrical transmissions between two or more terminals and/or systems.

DATA COMMUNICATIONS SYSTEM:

A data communications system which consists of two or more computer systems that are linked by means of a communication line(s) for the purpose of data exchange.

DATA LINK

A composite of computer systems and the interconnecting network, together with control procedures, operating in a particular mode that permits information to be exchanged between these systems.

DATA SET:

A device that provides an interface between a data communications system and a data communications line. Most data communication lines use a modulated carrier for data transfer. The data-set provides the logic to modify the data that is being received or transmitted between terminals and the system.

DATA SET READY:

A signal provided by the Data Set.

DATA TRANSMISSION:

Data transmission consists of transmitting data between two or more computer systems in a system network.

DISCONNECT (DEOT):

The term (and mnemonic) for the "DLE EOT" communication control sequence which is used to signal that a "disconnect" of a switched circuit must be initiated.

DIAL-OUT:

The use of an Auto Call Unit to establish a "switched" connection between two (2) computers.

DIAL-UP:

The use of a dial or touchtone data set to establish a "switched" connection between two (2) computers.

DISK CARTRIDGE:

A secondary data storage device much the same as a "disk pack" but usually smaller in size. It can be moved online or offline.

TERM : DEFINITION

**DISK DIRECTORY:**

A disk-resident table that contains the name and type of file, together with a pointer to the disk file header or sub-directory for all permanent files which reside on disk.

**DISK PACK:**

A secondary data storage device that can be moved online or offline.

**DISK-PACK-IDENTIFIER (dp-id):**

An identifier which is the name assigned to a disk pack or disk cartridge.

**EXTENDED BINARY CODED DECIMAL INTERCHANGE CODE (EBCDIC):**

A code consisting of 8-bits which is used to represent 256 unique letters, numbers and special characters.

**ERROR:**

Any received character or sequence of characters that does not conform to those transmitted, are considered to be in error.

**ERROR RECOVERY PROCEDURES:**

These are data communication control procedures used to restore normal operations to a data link after an error condition has occurred.

**FAMILY NAME:**

A identifier used as a file name, or the name assigned to identify a main file with sub-directory entries.

**FILE SECURITY:**

The procedures or special devices used to prevent access to or use of data or programs without authorization.

**FILE IDENTIFIER:**

All disk file-identifiers used on the system must be unique to prevent duplicate file-names. A file identifier can be comprised of any combination of the following file-identifier options:

- . file-identifier
- . family-name/file-identifier
- . dp-id/family-name/file-identifier

**HALF-DUPLEX CHANNEL:**

This is a communication channel in which the signaling may be in either direction, but not in both directions simultaneously.

TERM : DEFINITION

**HALF-DUPLEX TRANSMISSION:**

This is a type of data transmission where information is sent in only one direction at a time. It cannot send data in both directions simultaneously.

**HARD COPY:**

Generally, the printed copy of some material in a readable form is considered to be "hard copy". This would include computer printed media such as Reports, Listings and documents like this Software Operational Guide.

**IDENTIFIER:**

A word consisting of from one to seventeen numeric, alphabetic, or special characters in combination.

**INIEGER:**

A whole number as opposed to a fraction; that is, a number that contains the unit of (1) an exact number of times.

**INTER-SYSTEM COMMUNICATIONS:**

The ability for a terminal or program in one system to communicate with a program in another system.

**INTRA-SYSTEM COMMUNICATIONS:**

The restricted ability for a terminal or program in a system to communicate with another program in the same system.

**JOB**

A "job" to this system is any program that is executable under the control of the MCP; SYCOM is a job.

**JOB NUMBER:**

A unique number assigned by the system's MCP to each job that is executed under its control.

**JOB SPAWNING:**

The ability for a user to execute jobs under program control and maintain an adequate level of control over those jobs during their execution.

**KEYBOARD:**

A device for the encoding of data by key depression, which causes the generation of the selected code element.

**LEASED LINE:**

A data transmission line reserved for the exclusive use of a customer (data center).

TERM : DEFINITION

**LIKE SYSTEM:**

A "like system" is a system with the physical characteristics similar to those of a target system. For a B1800/B1700 system, a "like" system would be either a B1800 or B1700 system.

**LINE:**

A data communications line linking two (2) computers or a computer and its associated terminals.

**LINE ADAPTER:**

A line adapter consists hardware and/or software logic used to interface a data-set or communications line into a multi-line or single-line I/O control unit.

**LINE SWITCHING:**

The technique of temporarily connecting two communications lines together so that two computers and/or a computer and its associated terminals can exchange data.

**LOG-OFF:**

The procedure by which a user ends a data exchange session either between two computers or between a computer and a terminal.

**LOG-ON:**

The procedure by which a user establishes a data exchange session between two computers or a computer and a terminal.

**LOGICAL STATION NUMBER (LSN):**

The identification number assigned by the Network Controller to a given station (terminal). The Logical Station Number has no relationship to file names, but is determined by the order in which the stations are defined in the Network Controller.

**MASTER CONTROL PROGRAM (MCP):**

The Master Control Program is the master "control state" program (operating system) used in all B1800/B1700 series computers for system control.

**MESSAGE:**

A sequence of characters arranged in a form suitable for the purpose of conveying information. A message contains the data to be conveyed (the text) and may, in addition, contain communications characters to aid in the routing or handling of the message. MESSAGE: A sequence of characters arranged in a form suitable

**TERM : DEFINITION**

**MESSAGE FORMAT:**

A message format defines the placement of message elements within the message text. These elements would be such items as; message header, control characters, and text data.

**MESSAGE HEADER:**

Every message moved between a terminal and the SMCS, or an application program for that matter, is prefixed with a "message header". This header contains the station attributes necessary to identify the station when the time comes to return a reply message.

**MESSAGE NUMBERING:**

Message numbering is the sequencing of the messages transmitted by the communications system to ensure that all messages sent are in turn received.

**MODEM:**

Same as "Data Set".

**MULTIFILE IDENTIFIER (MF-ID):**

A generalization for a group of files with the same "family name" but with different file names.

**MULTI-LINE CONTROL (MLC):**

The (MLC) is an I/O control device that provides the functional control between computer hardware and the line adapters. The control is an integral part of a multi-line communications sub-system.

**NETWORK:**

A number of data communication lines connecting the User System with remote work stations (terminals).

**NETWORK CONTROLLER:**

The NDL generated data communication program used by SYCOM to interface with the work stations (terminals) and other systems.

**NETWORK DEFINITION LANGUAGE (NDL):**

A descriptive, free-form language, for defining and implementing a data communications network. The NDL compiler analyzes the input statements and generates a custom Network Controller.

**NORMAL STATE:**

The term "normal state" for this system, generally refers to a program which has no facilities to seize control of a system's processor; as opposed to "control state". (e.g., SYCOM is a normal state program.)

**TERM : DEFINITION**

**ODT:**

The currently accepted term for the B1800/B1700 systems' "Operator Display Terminal" (usually a video display terminal). Usage in this document will be the abbreviation "User ODT".

**OFFLINE:**

There are two facets to this term for the SYCOM system; 1) a system is "offline" when no communications link currently exists between it and a host system, and 2) a system's terminal, ODT, or program is "offline" when it is not logically connected to an application program in a host system.

**ONLINE:**

There are two meanings for this term in a SYCOM system; 1) a system is "online" when a communications link currently exists between it and a host system that provides processing services, and 2) a terminal, ODT, or program in a User System is "online" when it is logically connected to an application program in a host system. (The system must be online in order for a terminal, ODT, or program to be online.)

**PARTICIPATE:**

Message Control System (SMCS) which allows it to "participate" in all input/output message traffic between the Network Controller and an application program. In reference to a SYCOM system, the SMCS is not permitted to participate in SYCOM's message traffic.

**PASSWORD:**

A unique string of characters that a program, computer operator, or terminal user must supply to meet security requirements before gaining access to data/files. The password is confidential as opposed to the "usercode" which is not.

**POINT-TO-POINT CONNECTION:**

This is a configuration in which a network connection is established between two, and only two, computers. The connection may include a switching facility.

**PRIVATE LINE:**

This would be a data communications line which has been furnished to the user for exclusive use.

**TERM : DEFINITION**

**PSEUDO CARD READER:**

In a B1800/B1700 system, a pseudo card reader is in fact a disk file. The disk file may be created via cards, program output, or through a terminal. When a program requires an input card file, the pseudo card reader can be substituted; as that file can be appropriately labeled.

**QUEUE:**

A logical collection of messages awaiting transmission or processing.

**REMOTE JOB: 1**

Submission of control statements and data from a remote station (terminal/computer), causing the job described to be scheduled and executed at another system.

**REMOTE JOB ENTRY (RJE):**

The Remote Job Entry program is a method by which a remote computer executes or complies programs sent to it by other computers; which are considered terminal devices.

**REMOTE SUPERVISORY CONTROL STATION (RSC):**

This is an input/output device located on a User System (remote to a Server System) which provides the operator with the ability to control jobs and to request information from the Server System.

**RJE/TERMINAL:**

The official name for the software system from which SYCOM derived its basic capabilities.

**RJE STATION:**

For this system, the terms "RJE Station" and "User System" are equivalent. The new term is used to denote a system with many additional capabilities and potential, therefore, the term "RJE Station" is no longer appropriate.

**RELATIVE STATION NUMBER (RSN):**

Is the position within the Remote File family; assigned at generation time.

**SEMANTICS:**

The relationship between symbols and their meanings.

**SERVER SYSTEM:**

A Server System is a passive system which provides computing services to the "User System"; locally or remotely.

TERM : DEFINITION

**SESSION:**

A period of time during which a user engages in dialog between a terminal and a computer or between two computers. (The elapsed time between when the user logs-on the system and when the log-off occurs.

**SESSION NUMBER:**

A unique number assigned by a criginating system to each session.

**SINGLE LINE CONTROL (SLC)**

This is an I/O control device that provides the functional control between a computers main frame and a line adapter. This control is an integral part of a single line communications sub-system.

**SIIE ID:**

A unique 17-character identifier sent in a special message format to a server system prior to being logged-on to provide a positive means of identification, independent of any other system security.

**SPO LOG:**

A disk file that contains all input/output messages sent to and received from the User ODT.

**SPOOLING:**

The reading and writing of input/output data streams on secondary storage devices, in a format convenient for later output operations and/or processing.

**SIATION:**

Same as "TERMINAL".

**SWITCHED LINE:**

A communications line with which a temporary connection between a User System and a Server System is established by dialing.

**SWITCHING:**

This is an operation involving the inter-connecting of circuits in order to establish a temporary communications link between two or more systems.

TERM : DEFINITION

**SYNCHRONOUS TRANSMISSION:**

A data transmission process in which the sending and receiving devices are operating continuously at substantially the same frequency and are maintained, by means of correction, in a desired phase relationship.

**SYSTEM DISK:**

A disk pack or disk cartridge that is initialized as a system pack. A system pack is under control of the MCP and one or more must be present on a system for the MCP to function. Head-per-track disk is always considered system disk.

**TERMINAL:**

This is an input/output device designed to receive and send source data in an environment associated with the job to be performed. Further, it is capable of sending entries to, and obtaining output from, the system of which it is a part.

**TEXT:**

A sequence of characters forming the information to be conveyed between systems and/or terminals and systems.

**TRANSMISSION:**

Same as Data Transmission.

**TRANSMISSION BLOCK:**

A transmission block may contain all or part of a message block and is independent of the message(s) from which its characters are taken.

**TRANSPARENT MODE:**

An operational mode in which all coded combinations of eight-bit characters are allowed as message text characters.

**UNIT DEVICE IDENTIFIER (udi):**

The mnemonic identifier assigned to each peripheral device attached to the system.

**USER ODT:**

The User System's Operator Display Terminal (ODT). 1

**USER SYSTEM:**

A User System is the active participant system providing local service to the user wishing to utilize another computing resource, locally or remotely.

TERM : DEFINITION

USERCODE

A symbol which consists of from one to eight characters to identify the system user.

VIRTUAL TERMINAL:

A logical communications capability which permits a terminal, ODT, or program in one system to "interact" with an application program in another system.

VDI:

Video Display Terminal. A video type work station (terminal) such as the TD820 and TD830 terminal devices.

WAITING SCHEDULE:

Contains the names and run-time attributes of jobs that are waiting to be placed into the Active Schedule.

ZIP

A method by which a program such as SYCOM, communicates with the MCP for certain specific MCP functions.

ACRONYMS

ACU - Auto Call Unit  
AP - Application Program  
ASCII - American Standard Code for Information Interchange  
BCC - Block Check Character  
BOJ - Beginning Of Job  
bpi - Bits Per Second  
D/S - Data Set  
DDD - Direct Distance Dialing  
dp-id - Disk Pack Identifier  
EOJ - End Of Job  
I/O - Input/Output  
lsn - Logical Station Number

MCP - Master Control Program  
NC - Network Controller  
NDL - Network Definition Language  
NIF - Network Information File  
ODT - Operator's Display Terminal  
RJE - Remote Job Entry  
RSN - Relative Station Number  
SMCS - Supervisory Message Control System  
SOG - Software Operational Guide  
SYCOM - Systems Communications Module  
udi - Unit Device Identifier  
VDT - Video Display Terminal

INDEX

(SYSTEM)/USERCODE File 4-36

ABORT 5-5, A-3  
ACRONYMS 9-13  
ADD 5-8  
APPENDIX B: CHARACTER SET B-1  
APPENDIX C: COMMAND SUMMARY C-1  
APPENDIX D: SYNTACTICAL KEYWORD SUMMARY D-1  
APPENDIX E: COMMUNICATIONS CONTROL MESSAGES E-1  
APPENDIX F: RESTRICTED MCP COMMAND SUMMARY F-1  
APPENDIX G: PARAMETERS FILE ATTRIBUTES G-1  
APPENDIX H: SUMMARY OF FILES H-1  
APPENDIX I: SAMPLE APPLICATION PROGRAMS I-1  
APPLICATION PROGRAMS 3-11  
Assigning Program Priorities 6-10  
ASSOCIATED DOCUMENTATION 2-5  
  
BASIC DATA COMMUNICATION MESSAGES E-2  
BASIC FEATURES 2-2  
BUFFER 5-13, A-3  
BUFFER Parameter 6-15  
BYE 5-14  
  
CARD READER/PUNCH MESSAGE E-3  
CL A-3  
CL (Clear Device) 5-15  
COMMAND INFORMATIONAL/ERROR MESSAGES A-2  
COMMAND RELATED ERROR MESSAGES 5-4  
COMMAND SYNTAX CONVENTIONS 5-2  
Comments 5-3  
Common System Operations 6-13  
COMMUNICATIONS PROCEDURE 3-5  
COMPILATION PROCESS 8-2  
CONCEPTS AND FACILITIES 3-1  
CONFIGURATIONS AND SOFTWARE ASSOCIATIONS 4-20  
CONNECT 5-16, A-3  
CONVERSATIONAL COMMUNICATION PROCEDURE 4-17  
COPY 5-17, A-4  
CP NR (Card Punch Not Ready) 5-22  
Creating the SYCOM Parameters File 6-5  
Creation of the (SYSTEM)/USERCODE File 6-10  
  
DEBUG 5-23, A-7  
DEBUG Parameter 6-15  
DEFAULTPSEUDOSIGNAL 6-15  
DETERMINING THE SPECIFICATIONS 8-1  
DISCONNECT 5-24, A-8

END Parameter 6-15  
Entering the Parameters 6-14  
Entering the Parameters Through the ODT 6-9  
EOJ 5-25, A-8  
ERROR DETECTION AND RECOVERY 2-4  
Example Interactive Terminal Session 7-5  
Execution of SYCOM; With SMCS 6-13  
Execution of SYCOM; Without SMCS 6-11

FETCH 5-26  
FETCH (also see COPY) A-8  
FILE SEPARATOR MESSAGES E-5  
FILE TRANSFER MESSAGE E-4  
FILE TRANSFERS 4-3  
FOR THE LEVEL 1 SYSTEM 4-26  
FOR THE LEVEL 2 SYSTEM 4-30  
FOR THE LEVEL 3 SYSTEM 4-32  
FUNCTIONAL DESCRIPTION 4-1  
FUNCTIONS PERFORMED 2-2

GENERAL 2-1, 3-1, 4-1, 5-1, 6-1, 8-1, A-1, E-1  
GENERATING NETWORK CONTROLLERS 8-1  
GLOSSARY OF TERMS AND ACRONYMS 9-1  
GROUPSIZE 5-30, 6-16

HARDWARE REQUIREMENTS 3-8

INBUFFERS 6-16  
INPUT COMMANDS 5-1  
INTRODUCTION TO SYCOM 2-1

LEVEL 1 SYSTEM 4-20  
LEVEL 2 SYSTEM 4-22  
LEVEL 3 SYSTEM 4-24  
LIMITATIONS 2-4  
LINE PRINTER MESSAGE E-3  
LINE THROUGHPUT 2-3  
LIST 5-31, A-8  
LOG 5-32  
LP 5-33

MAINTAINING INTEGRITY OF FILE CHARACTERISTICS 4-4  
MANUAL OBJECTIVES AND READER ASSUMPTIONS 1-1  
MASTER CONTROL PROGRAM (MCP) 3-9  
MASTER CONTROL PROGRAM (MCP) FILES 4-36  
MASTER CONTROL PROGRAM MESSAGES A-1  
MEMORY REQUIREMENTS 3-8  
MESSAGE FORMATS E-2  
MESSAGE PATH ASSIGNMENTS 4-26  
MESSAGE TYPES E-1

NETWORK CONTROL 4-17  
NETWORK CONTROLLER 4-19

NETWORK CONTROLLER ERROR MESSAGES A-1  
NETWORK CONTROLLER FILES 4-36  
NETWORK CONTROLLER PROGRAM 3-10  
Network Information (NIF) File 4-36  
NOFS Parameter 6-16  
NOFTP Parameter 6-16  
NOPROGRAM 6-16  
Notation used in model format 5-2  
NOTICE 2-5  
NOTRACE 6-17  
NOWAIT 6-17

ODT MESSAGES E-3  
OFFLINE 5-34, A-9  
ONLINE 5-35  
OPERATING PROCEDURES 6-1  
OPERATIONAL PROCEDURES 8-1  
OPERATOR COMMANDS 5-1  
OUTBUFFERS 6-17  
OUTPUT AND ERROR MESSAGES A-1

Peripheral Files 4-36  
PHONE 5-36, A-9  
POINT-TO-POINT CONTENTION PROCEDURE 4-18  
POINT-TO-POINT NON-SWITCHED NETWORK 3-4  
POINT-TO-POINT SWITCHED NETWORK 3-3  
PREFACE 1-1  
PRELIMINARY PROCEDURES 6-1  
PREPARATION OF INPUT 8-1  
Preparing the NEWPARAMS Card File 6-7  
Preparing the NEWPARAMS Disk File 6-7  
PRINT THE USER/NCNIF FILE 8-2  
PRINTING THE TRACE FILE 7-3  
PROGRAM 5-38  
PROGRAM Parameter 6-17  
Program Switches 6-2  
PROGRAM-to-PROGRAM (Server System) I-2  
PROGRAM-to-PROGRAM (User System) I-1  
PROGRAM-TO-PROGRAM COMMUNICATIONS 4-15  
PROGRAM-to-PROGRAM MESSAGES E-6  
PROGRAM-TO-SYCOM COMMUNICATION 4-13  
PROGRAM-to-SYCOM MESSAGES E-6  
Punctuation Rules 5-3  
PUT 5-39  
PUT (also see COPY) A-9

QS A-9  
QS (Queue Size) 5-43  
QUEUE DEPTH CONTROL 4-35  
Queue Files 4-38  
QUEUESIZE Parameter 6-17

REACTIONARY HINTS AND POINTERS 7-3  
READ 5-44, A-10

RECORD FORMAT G-1  
Remote Files 4-37  
REMOTE SUPERVISORY CONSOLE (RSC) 4-11  
RESTART 5-46, A-10  
RESTRICTIONS 4-38  
RY A-10  
RY (Ready Device) 5-48

SAMPLE NDL SOURCE FILE (LOCAL USER SYSTEM) 8-3  
SAMPLE NDL SOURCE FILE (REMOTE SERVER SYSTEM) 8-6  
SCOPE AND OBJECTIVES 2-1  
Setup MCP Program Name Table 6-10  
SIGNAL 5-49  
SIZE 5-50, A-10  
SMCS Command String from Server-System 5-60  
SMCS Command String from User-System 5-58  
SMCS MCP Command String from Server-System 5-61  
SMCS MCP Command String from User-System 5-59  
SMCS Signal Characters 6-14  
SOFTWARE OPTIONS 3-10  
SOFTWARE REQUIREMENTS 3-9  
STANDARD RJE FUNCTIONS 4-1  
STOP 5-51, A-11  
STRUCTURE OF THE DOCUMENT 1-1  
SUPERVISORY MESSAGE CONTROL SYSTEM (SMCS) 3-10  
SV A-11  
SV (Save Device) 5-52  
SYCOM AND ITS ENVIRONMENT 3-8  
SYCOM Commands from the ODT 7-1  
SYCOM ERROR MESSAGES A-2  
SYCOM FILES 4-36  
SYCOM PROGRAM SHUTDOWN 7-2  
SYSTEM COMMUNICATIONS 3-2  
SYSTEM CONDITION MESSAGES A-14  
SYSTEM CONFIGURATION 3-1  
SYSTEM CONTROL MESSAGE E-2  
SYSTEM FATAL MESSAGES A-17  
SYSTEM FILES 4-35  
SYSTEM OPERATING PROCEDURES 6-1  
SYSTEM PARAMETER ERROR MESSAGES A-12  
SYSTEM WARNING MESSAGES A-12  
SYSTEM/MAKEUSER PROGRAM 3-9  
SYSTEMID 5-53  
SYSTEMID Parameter 6-18

TERMINAL FUNCTIONS 7-5  
Terminal Operations (SMCS Controlled) 7-8  
TERMINAL OPERATIONS (SYCOM CONTROLLED) 7-5  
TERMINAL PROCEDURES 7-5  
TERMINAL-TO-PROGRAM COMMUNICATIONS 4-6  
TERMS 9-1  
The Server System 4-10  
The User System 4-9  
TIMEOUTS Parameter 6-18

BURROUGHS CORPORATION  
COMPUTER SYSTEMS GROUP  
SANTA BARBARA PLANT

IX-5  
COMPANY CONFIDENTIAL  
B1800/B1700 SYCOM  
P. S. 2219 0458 (A)

TRACE 5-54, A-11  
TRACE Parameter 6-18  
TRACEAREAS 6-18  
TRACEBA Parameter 6-18  
TRACEFB Parameter 6-19  
TRANSMISSION OF PACKED DATA 4-19

USER 5-55, A-11  
USER SYSTEM - SERVER SYSTEM RELATIONSHIP 3-6

VIRTUAL TERMINAL MESSAGE E-4  
VIRTUAL TERMINALS 4-5

WAIT 5-56  
WAIT Parameter 6-19  
WHAT 5-57  
Workfile Naming Conventions 6-4