BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

Software Design Specification

TABLE OF CONTENTS

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

SCOPE

This document describes the Burroughs Systems Communications
Module and establishes standards and guidelines for the system.
The data communication procedures used are the Burroughs Data
Communications Standards as defined and published by the Data
Communications Committee and Corporate Engineering. Other areas
of SYCOM, such as record formats and supervisory messages are
standardized in this document.

In addition to RJE, this protocol includes the features of file
transfer,   program-to-program communication and the issuance of
SPO commands between two computers (or from a terminal to a
computer).   These latter features are symmetric;   they may be
invoked between any two systems without regard to "host" vs.
"terminal" status.

The functions that might be supported by a program implementing
this protocol fall into three categories: host RJE, terminal
RJE,   and file transfer.   This protocol specifically allows
implementation of a subset; any one or any combination of these
functions is allowed.

DEFINITION OF SYCOM

The purpose of SYCOM is to increase the use of data processing systems located some distance from each other. SYCOM includes remote job entry and file transfer capabilities. In remote job entry, jobs are submitted through a remote terminal and transmitted via a data communications line into the job stream of the host system. After processing, the job's printed and punched output is transmitted back to the terminal via the communications line. File transfer is the movement of files between two systems which can be either "terminal" systems or "host" systems.

Since some of the functions of this protocol allow interaction of 2 peer-level entities, the terms "terminal" and "host", although still used, are less applicable. Most applicable are the notions of "user" system and "server" system, as used in the ARPA Network. The user system is the active party with the human user wishing to utilize a computing resource, and the server system is the passive party supplying the computing service.

The basic functions of SYCOM are:

1. Introduction of programs from a remote input device for compilation and/or execution.

2. Introduction of data from a remote input device for processing by the host system.

3. Dispersing of data and/or programs produced by the host system to the remote device(s).

4. Monitoring and controlling of programs on the host system via a remote supervisory console.

5. Transfer of files from one system to another.

6. Support of communication between programs executing on the two systems.

An implementation of this protocol should include a system-dependent way for a user program to initiate any SYCOM function.

Since SYCOM applications imply the transmission of large batches of data between the two systems, line throughput maximization, error detection, and error recovery are of prime importance. This section discusses those topics and the procedures and conventions used in SYCOM applications.

LINE THROUGHPUT

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

1. Specifically acknowledges a message with either a message or a numbered acknowledgement (#ACK). This reduces both line time and line turnaround per message.

2. Blocks records (that is, sends multiple records per message).

3. Compacts and expands blanks. It is common on card input and printer output to have a large percentage of blanks imbedded in the data. Eliminating the transmission of blanks can significantly improve throughput.

4. Sends messages in one direction while error recovery proceeds in the opposite direction through the unique capability for inclusion of a specific acknowledgement to a received message within a transmitted message.

ERROR DETECTION AND RECOVERY

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

1. A specific numbered acknowledgement (either in a message or in a #ACK) guards against the possibility of lost messages by providing an unambiguous acknowledgement to the transmitted message.

2. The use of at least modulo 2 transmission numbering facilitates the detection of duplicates.

3. Dynamic reduction of the transmission message length provides the potential for reducing error rates and retransmissions on bad circuits (error rates per message increase as the message size increases).

4. The convention of different timeout values for central and remote system alleviates line collision problems.

5. The message transfer procedure eliminates the ambiguous use of certain ASCII control characters (for example, ENQ is used only during establishment, nowhere else).

6. Error detection is provided through the use of vertical and horizontal parity in SYCOM.

BURROUGHS CORPORATION   COMPANY CONFIDENTIAL
COMPUTER SYSTEMS GROUP   B1800/B1700 SYCOM
SANTA BARBARA PLANT   SDS 2222 2590 (02)

DATA COMMUNICATIONS PROCEDURE OF SYCOM

The point-to-point communications message transfer procedure
allows the reversal of direction of message transfer without
separate establishment and termination procedures. The essential
characteristic of this procedure is the ability to specifically
acknowledge a previous transmission via the use of a character
included within the header of the message. This character will
hereinafter be referred to as the ARM (Acknowledge Received
Message) character and will be defined to have the following
values:

1. ARM is the transmission number of the last validly
    received message, or NUL, immediately following
    establishment.

2. The ARM character immediately follows the SOH.

The validity of the ARM character will not be considered apart
from the validity of the message (that is, if a message has a
parity error, the ARM will be ignored).

The usefulness of the ARM is twofold: First, it provides a
specific acknowledgement to a previous transmission because of
its "numbered ACK" facility. Secondly, when a receiving station
receives a message that it cannot accept because of a no-buffer
or not-ready condition, the ARM allows the facility to implicitly
NAK that message, while at the same time being able to transmit a
new message and thereby possibly freeing buffers and/or conveying
in the test the not-ready or buffer-full condition.

SYCOM HOST SYSTEMS AND REMOTE TERMINALS

The following systems have implemented or plan to implement the
following packages:

	hRJE	tRJE	FTP	term
B7700/B7800	C		P	P
B6700/B6800	C		P	P
Medium systems	C			
B1800/B1700	C	C	C	C
B800		C	P	P
B80		C	P	P
B771/B772		C		
TC5100		C		
TC3500		C		
DC1000		C		

    Where hRJE = host RJE
           tRJE = terminal RJE
           FTP = file transfer
           term = terminal-to-program and program-to-program

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

interface
C = currently available
P = planned for implementation

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

SYSTEMS COMMUNICATIONS STANDARDS

SYCOM

SYCOM defines the standard functional characteristics for a nontransparent half-duplex mode systems communications mechanism.

SYCOM DATA COMMUNICATIONS STANDARD

This section defines the conversational SYCOM link procedure for operation over directly connected, leased, or switched lines. This SYCOM link procedure is a subset of the Burroughs Conversational Point-to-Point Line Procedure as cefined in Burroughs Corporate Data Communications Standards, Document 1284 9006.

CONVERSATIONAL COMMUNICATIONS PROCEDURES

Figure 3-1 defines the basic conversational message transfer procedure. Conversational procedures allow reversals of direction of message transfer without separate establishment and termination procedures. This figure applies to a point-to-point circuit, either switched or dedicated. The establishment and termination procedures of figure 3-2 apply. An alternate timeout and recovery procedure is used whereby one station provides special timeout and recovery services for the other station.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

FROM ESTABLISHMENT WITH STATION A AS MASTER

NOTES

1. [ ] INDICATES OPTION THAT MAY BE EXCERCISED FOR SPECIFIC SYSTEMS.

2. REFER TO "Message Transfer"

3. REFER TO "Retransmission of Messages"

4. REFER TO "Parity or Block-Check Error Handling"

5. REFER TO "Message Idle State"

6. REFER TO "Normal Termination"

7. TEXT LENGTH IS VARIABLE.

SOH
ARM
XM#
STX
DA1
DA2
.
.
TEXT
.
.
ETX
BCC

(NOTES 2 & 7)

NEXT MESSAGE
REPEAT
[REPEAT] (NOTE 1)

FROM ESTABLISHMENT WITH STATION B AS MASTER

INR   [PREFIX] NAK   [PREFIX] ACK

(NOTE 3)

(NOTES 1 & 4)

(NOTES 1 & 5)

SOH
ARM
XM#
STX
DA1
DA2
.
.
TEXT
.
.
ETX
BCC

[REPEAT] (NOTE 1)
REPEAT
NEXT MESSAGE

[PREFIX] ACK   [PREFIX] NAK   INR

(NOTE 3)

(NOTES 1 & 4)

(NOTES 1 & 5)

TO ERROR RECOVERY

TO NORMAL TERMINATION (NOTE 6)

TO NORMAL TERMINATION (NOTE 6)

TO ERROR RECOVERY

Station A   Station B

FIGURE 3-1   CONVERSATIONAL COMMUNICATIONS PROCEDURES

Message Transfer

Message blocking is prohibited. Messages are delineated by special characters. A message begins with a SOH character and ends with a BCC character. The message stream is as follows: SOH, ARM, XM#, STX, DA1, DA2, the text of the message, ETX, and BCC.

Retransmission of Messages

If the master (transmitting) station does not receive a reply to its last transmission within some period of time, it may retransmit the unacknowledged message.

Parity or Block-Check Error Handling

If the slave station receives a recognizable message, but it contains character parity error(s) or a block check error, it may reply NAK. The master station should retransmit the message. This may be repeated 'n' times (n is user-specified and at least one), and if the error persists, either station may exit to error recovery.

Message Idle State

If the slave station receives a good message, but has no traffic to send, it responds ACK. If the master station has more traffic it sends another message. If the master station has no more traffic, it proceeds to the message idle state.

Normal Termination

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

POINT-TO-POINT CONTENTION PROCEDURES

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 stations are in an idle status (neither master nor slave), and that either station may bid for master status by sending the DLE ENQ sequence. See figure 3-2.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT
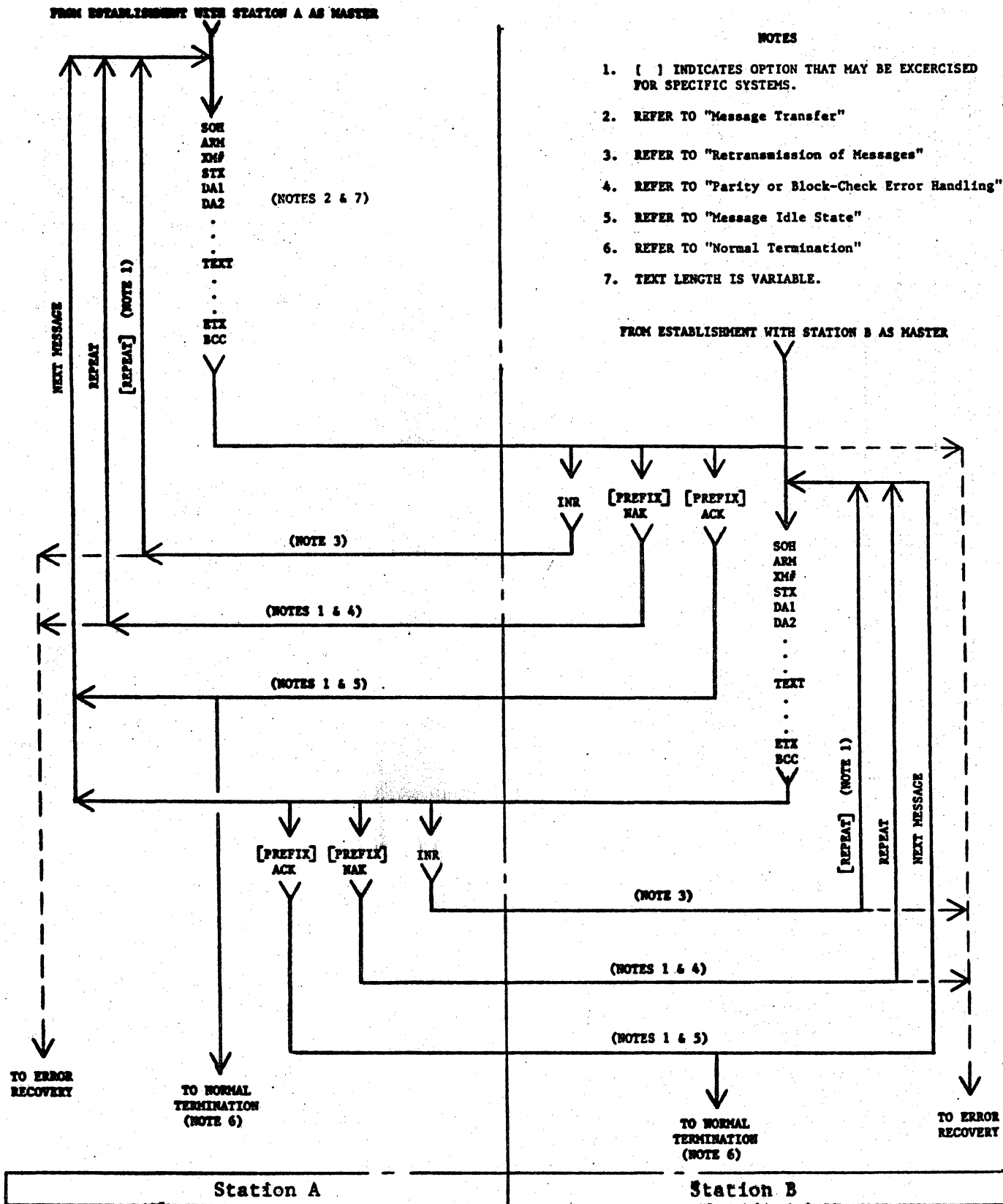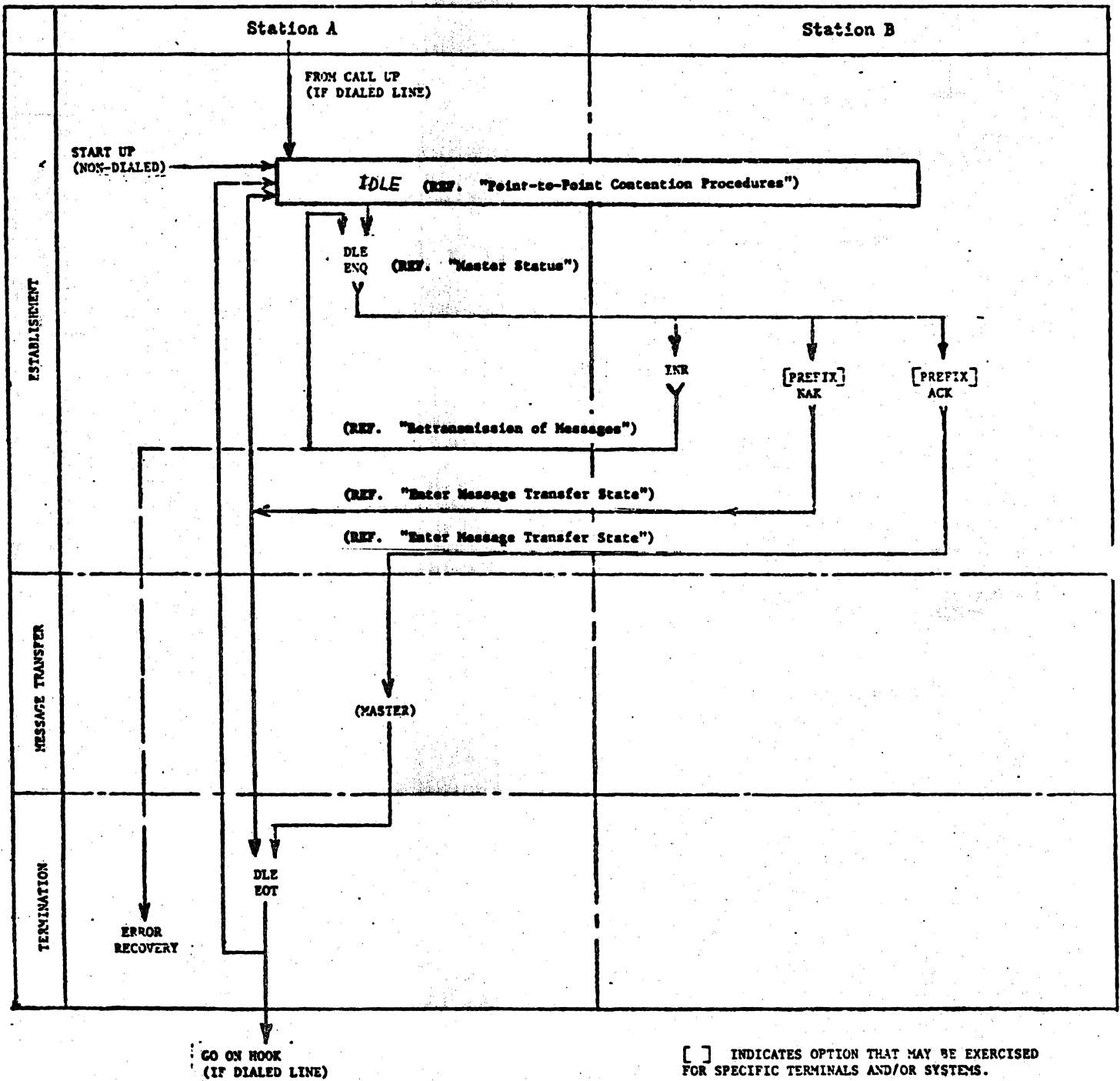
COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

Figure 3-2   Point-to-Point Contention Procedures

Master Status

If either station in the idle state wishes to enter message transfer, it may bid for master status by sending DLE ENQ. If the same station had just terminated message transfer by sending DLE EOT, it is recommended that there be a delay before this station sends DLE ENQ again.

Enter Message Transfer State

If the station bidding for master status receives an ACK (indicating ready to receive) it may enter message transfer. If the response is NAK (not ready to receive), the DLE ENQ may be repeated up to a retry limit of "n", (n is user-specified and at least one), and if no valid response is received, go to error recovery. If the station bidding for master status receives DLE ENQ, indicating that the other station is bidding at the same time, it shall abandon its own bid and reply in accordance with its readiness to receive. In order to avoid continuing line collision when both stations are bidding at the same time, the no-response timeout values should be of different length.

Termination

DLE EOT is transmitted to indicate the end of this session and, if switched lines, to disconnect the modes by going on hook.

FIGURE 3-3   SYCOM FLOW CHART

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

SYCOM PROCEDURE

The SYCOM flow chart provides a logic diagram of the detailed
SYCOM procedure (see figure 3-3). Information and definitions
for understanding the procedures are as follows:

1. The SYCOM procedure allows reversals of direction of
   message transfer without separate establishment and
   termination procedures. A message can be positively or
   negatively acknowledged with another message. This is
   accomplished by the use of the ARM (Acknowledge Received
   Message) character (defined below) contained within the
   message header.

2. Message blocking is prohibited. Each message is one
   block long, beginning with an SOH and ending with ETX
   followed by the BCC. The message header contains four
   characters defined as: SOH, ARM, XM#, and STX.

3. The ARM character indicates ACK or NAK of the previous
   message with respect to the validity or invalidity of
   that message. It is included:

   a. within the header of each message.

   b. as a prefix to an ACK character.

   The ARM will be initialized to a NUL character (00)
   during the establishment phase of the procedure. When a
   valid message is received, the XM# will be saved and used
   as the next ARM character transmitted. It will be set to
   the NUL character value until a valid message has been
   received.

   When an ARM sequence (ARM-ACK or SOH-ARM) is received, it
   will be compared with the last XM# sent and, if equal,
   the last message will be positively acknowledged. If the
   sequences do not agree, the last message has been
   negatively acknowledged and will be retransmitted.

   The validity of the ARM character will be checked only if
   a valid header and message is received, that is, the ARM
   character will be assumed to be a NAK unless received in
   the header of a valid message or in a valid ARM-ACK
   reply.

4. The XM# is a one-character field. The recommended modulo
   2 character sequence is F/U for the host, and 5/L for the
   terminal. The XM#'s received will be checked against the
   previous XM# received and, if equal, a duplicate text is
   assumed and the text is ignored. It shall hereby be
   noted that the NUL character (00) must not be used as a

transmission number, after the first transmission.

5. An invalid character punched in the first column of an 80-column card (or a question mark character in the first column of a 96-column card) is replaced by a NUL character (00) prior to transmission.

6. Recommended timeout values for Invalid or No Response are 2.5 seconds and 3.0 seconds for terminal and host respectively.

CODES AND ERROR CHECKING

Character parity checking and longitudinal parity checking are used. An error in either check constitutes an erroneous message and will be negatively acknowledged. The character code used is the American Standard Code for Information Interchange (ASCII) ANSI X3.4 1967 (Appendix A). Each character consists of seven data bits and one parity bit. The parity checks are:

1. Longitudinal Redundancy Character (LRC). This is a redundant character added to the end of a transmission block for the purpose of error detection and control. The LRC is obtained as the binary sum (initialized to zero) without 'carry' on each of the bits of the characters following SOH, including ETX. The polynomial used in this "cyclic" accumulation is X to the power of eight plus one.

2. Character Parity. All characters, including the LRC, shall have the same parity. Synchronous transmission uses odd character parity and asynchronous transmission uses even character parity.

SYCOM RECORD (TEXT) FORMAT

The SYCOM record format is as follows:

```
    DA1 DA2 [CC] <text> [RS [CC] <text> ...]
or  DA1 DA2 [CC] FS 0 <text>
or  DA1 DA2 [CC] FS 1 <text>
or  DA1 DA2 [CC] FS 2 <text>
                        2
```

where:

DA1 and DA2 are device addresses indicating the source or intended destination of the text.

CC is used for control of special functions (only for the printer, not punch).

FS is the file separator.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

RS is the record separator.

<text> is a unit record, optionally modified by space compression and by deletion of trailing spaces. All text within a message is addressed to the same device.

DEVICE ADDRESSES (DA1, DA2)

DA1 and DA2 will be used for device addresses at the remote station. The convention used is:

DA1	DA2	Usage
0	0	System control
0	1	SPO
0	2	Card reader/card punch
0	3	Line printer
0	4	Data of a file transfer
0 through 8	5 through 1	Reserved for expansion
8	2	Packed punch
8 through 9	3 through 9	Reserved
a through z	A through Z	User host transmit, or server host receive, virtual station data.
A through Z	a through z	User host receive, or server host transmit, virtual station data.

See TERMINAL-TO-PROGRAM COMMUNICATION for more information on virtual station data. The device addresses are advisory only. The remote station may redirect a message to any appropriate device, based upon its own requirements.

Note that line printer data is allowed in either direction. When a system receives line printer data, it prints it on an available printer, independently of whether the system is "terminal" or "host".

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

DEVICE FUNCTIONAL CONTROL (CC)

A one-character field (CC) is used for control of special functions. The CC field is required and is valid only for messages addressed to a line printer. The CC field will always appear as the first character of a record when DA1=C and DA2=3, indicating that these records are destined for a line printer. The CC field controls the printer's vertical motion according to the following conventions (skip-to-channel is subject to hardware implementation):

CC	Function
0	Print, no spacing
1	Print, then single space
2	Print, then double space
3-9	Reserved for expansion
A	Skip to channel 1 - top of page
B	Skip to channel 2
C	Skip to channel 3
D	Skip to channel 4
E	Skip to channel 5
F	Skip to channel 6
G	Skip to channel 7
H	Skip to channel 8
I	Skip to channel 9
J	Skip to channel 10
K	Skip to channel 11
L	Skip to channel 12 - end of page

FILE SEPARATOR (FS)

A separate message containing a file-separator character (FS) is used to separate files. The character following the FS is either "0", "1", or "2", indicating beginning-of-header, beginning-of-file or end-of-file, respectively. The text portion of the beginning-of-header or beginning-of-file separator record contains the file name. This name may be used at the recipient's discretion.

Example: begin header: SOH ARM XM# STX C3A FS 0 <text> ETX
          begin file: SOH ARM XM# STX 03A FS 1 <text> ETX
          end file: SOH ARM XM# STX C3A FS 2 ETX

The file-separator record is sent from either system signifying that the beginning or end of file has occurred for this device, or to prefix a copy of the file's symbolic file header (see MAINTENANCE OF FILE CHARACTERISTICS for more information).

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

RECORD SEPARATOR (RS)

Records are separated by the use of the ASCII control character RS. Character RS is not an end-of-record or a beginning-of-record designator, but is only used between records as a separator when several unit records are combined into a single SYCOM message. SPO records are never combined.

SPACE COMPRESSION

A sequence of consecutive space characters, up to 99 in length, may be replaced by an ASCII escape control character (ESC) followed by a two-character decimal number containing a count of space characters replaced. Greater than 99 consecutive space characters will require two or more ESC sequences. Trailing blanks may be truncated. SPO or packed records are not space-compressed or truncated.

SYCOM SYSTEM MESSAGES

In addition to the standard datacomm control interface, systems in the SYCOM environment need to communicate certain information via system messages to insure orderly operation. System messages are those messages whose DA1 and DA2 characters are 00.

SYCOM system messages are two-character decimal codes, followed in some cases by additional data. In the following definitions the code will be listed followed by its definition. The definition will be enclosed in parentheses. Beneath the code and definition will be an explanation of the code. The following are currently defined:

01 (Logged On)   *<usercode>/<password>*   *?*

Sent by the host system to the remote terminal to inform the terminal that the operator using the terminal has been logged on.   *FFFF*

02DDDD (Buffer Size)

Sent from either system, requesting a change in the length of messages being assembled for transmission. DDDD is the requested message length in decimal notation, e.g., 0200 means a maximum message size of 200 characters. Initial default buffer size is 402 (number of characters between STX and ETX).

A Buffer Size Request message is always answered with a Buffer Size Reply (04) message. Upon being logged on, the terminal should send this message to the host specifying the size of the largest message it can receive. Thereafter, this message may be sent to request an increase or a decrease in

actual message size. A decrease might be requested to reduce
error rates on a marginal circuit.

It is important to note that the absolute maximum length of
messages which will be accepted on input is specified only by
the initial exchange of Buffer Size Request and Buffer Size
Reply messages. Subsequent requests exceeding the original
specification will result in message lengths conforming to
this specification.

The maximum buffer size is 9999.

03 (Logged Off)

Sent from the host to the remote terminal to indicate that
the terminal is no longer logged onto the host system.

04DDDD (Buffer Size Reply)

Sent from either system as a reply to Receive Buffer Size
request (02) message. DDDD is as defined for Buffer Size
Request. When replying to the initial Buffer Size Request
(after log on), the host system will specify the largest
message it can receive. After the initial Buffer Size
Request/Buffer Size Reply message exchange, either system
will reply to a Buffer Size Request message with a Buffer
Size Reply message containing the maximum length of message
that it will, henceforth, assemble. This length will be the
smaller of the following:

    1. The requested length.

    2. Its maximum buffer size.

05[DA1][DA2] (Device Not Ready - Suspend)

Sent from either system instructing the other to suspend
transmitting messages for DA1 and DA2. Indicates a temporary
condition at the device (e.g., line printer out of paper). A
Device Ready (07) system message should be sent to resume
transmissions when the condition is corrected. If the
condition is not corrected, a Device Not Ready Abort (06)
system message should be sent.

06[DA1][DA2] (Device Not Ready - Abort)

Sent from either system informing the other that an
uncorrectable condition exists preventing the processing of
additional transmissions for device DA1, DA2. The received
messages for that device are discarded. The transmitting
system must preserve the entire file.

07[DA1][DA2] (Device Ready)

Sent from either system:

1. After a Device Not Ready (05 or 06) system message to indicate that the condition has been corrected and the device is again ready to receive data, or

2. Whenever a device which was unavailable becomes available to receive data, or

3. Immediately after a Logged On (01) system message to indicate each device that is ready and available to receive data. (Optional) *otherwise no pur... ...* *Dick Holmes ...*

09CCCCLVVDDSSS (Station-id)

Sent from either system prior to logging on to provide a positive means of identification, independent of any security function. This message includes the sender's CPU model identification and an indication of which part(s) of the protocol he has implemented. A terminal must be prepared to accept this message.

"DD" is the number of characters in "SSS", which is the sender's station identifier (between 1 and 17 characters long, inclusive). "CCCC" is the sender's system type, left-justified and padded with blanks to 4 characters (e.g. "1700", "6700", "771 "). (This allows similar systems to identify each other and engage in specialized communication inappropriate between dissimilar systems.) "L" is a character that indicates which parts of the protocol the sender supports. It is constructed as follows: The high-order two bits are 01, the next two bits are 0, and each of the low-order 4 bits corresponds to one part of the protocol: hex 1 bit set means "can act as host", hex 2 means "can act as terminal", hex 4 means "can do file transfer", hex 8 means "can do terminal to program communication". (For example, an ASCII "A" would mean "can act as host", or ASCII "E" means "can act as host and can do file transfer".) "VV" is a 2-character decimal number indicating the version of the protocol implemented by the sender; this document is version "02".

```
01  LEVEL_RECORD,                        % length = 1 byte
    03   CONSTANT_BITS         BIT ( 4 ),  %          a(1)0100a
    03   TERMINAL_TO_PROGRAM   BIT ( 1 ),
    03   CAN_DO_FILE_TRANSFER  BIT ( 1 ),
    03   CAN_ACT_AS_TERMINAL   BIT ( 1 ),
    03   CAN_ACT_AS_HOST       BIT ( 1 );
```

This message may be sent by either system. On receipt, a host may initiate a log-on sequence, but a terminal may not. In the case of 2 hosts communicating with each other, either

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

one, both or neither may log on to the other. If both
systems send this message, they may not both specify the same
station identifier, since this is used as the "host name"
elsewhere.

10 (Put File)

Contains identification of the user originating the request
and filename and attributes of both source and destination
file (see "Format of Put/Fetch Messages" for complete
definition of format). The "Put" message elicits a response
of "Put reply".

11 (Fetch File)

Contains identification of the user originating the request
and filename and attributes of both source and destination
file (see "Format of Put/Fetch Messages" for complete
definition of format). Elicits a response (perhaps after a
time lag) of "Put File".

12TUUUUtext (User-destined Message)

"T" specifies whether destination is the SPO ("0"), a user
program ("1"), or a user at a terminal ("2"). "UUUU"
identifies the user program or terminal, in a system-defined
way. In any case, "text" is routed to the specified
destination. Used as answers to message type 13.

13TUUUUNNNN<filename> (List Fetches)

T and U have the same meaning as in message 12; NNNN is the
file number, if specified; otherwise blank. <filename> is
the filename if specified, otherwise it is blank (file name
can be "=").

15 (Close Request)

The recipient is to terminate the session in an orderly way,
after completing any file transfers it is currently
executing. The sender only sends this message when it is not
transmitting any files, and the receiver responds with a
Close Reply when it is not transmitting any files.

16 (Close Reply)

Sent in response to a Close message. On exchange of Close
replies, both systems are to terminate the session and break
the connection. If a switched line, both systems hang up;
if a leased line, it is an implementation decision whether to
re-establish or not.

NNNN

3-15

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

17CC <optional text> (Put Reply)

Sent in response to a Put message to indicate initial success
or failure of the file storage request.   "CC" is the outcome
code: "00" is success,   "01" is "failed for lack of
resources" (includes "packname not found", and "no more disk
space"),   "02" is "failed for security violation", "03" is
"failed for invalid filename syntax", "04" is "failed for
unimplemented device or file attributes", "05" means "this
system does not support file transfer" and "06" means file
already exists when "AS" requested.   Implementers are
encouraged to distinguish between various cases of these
errors,   and to recognize other errors;   the user should be
supplied with information as specific as possible. It is the
responsibility of the system doing the PUT, when it receives
a Put Reply with a failure code, to notify the user of the
error regardless of which system contains the user.   New
error codes will be added as needed.   The <optional text>
could be used to further explain (symbolically) the error
conditions.

18TUUUUNNNN<filename> (Abort Put)

Sent from either system to request that the specified PUT be
aborted, and that any portion of the file already written be
discarded. NNNN is the file number, if specified; otherwise
it is blank.   The <filename> is the file name, if specified;
otherwise it is blank (file name can be "=").   If no file is
specified (by number, name or "=") then the current PUT is
specified. If no PUT is in progress, this is a NOP.

19TUUUUNNNN<filename> (Abort Fetch)

Sent from either system to request that the specified FETCH
be aborted,   and to advise that any portion of the file
already written has been discarded. NNNN is the file number,
if specified;   otherwise it is blank. The <filename> is the
file name, if specified;   otherwise it is blank (file name
can be "=").   If no file is specified (by number, name or
"="), then the current FETCH is specified. If no FETCH is in
progress, this is a NOP.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

20VVcharacteristics (Connect)

Sent from user system to server system to request
establishment of a terminal-to-program connection. "VV" is
the virtual station address to be used for the proposed
connection, and is the user system's transmit address.
"characteristics" describe the characteristics of the real
terminal, to allow the server program to format messages
appropriately, if desired. The characteristics, taken from
Common NDL, are:

station ID	20 characters
line width	4 characters
CRT boolean	1 character
"0" = not CRT	
"1" = CRT	
MYUSE	1 character
"1" = input only	
"2" = output only	
"3" = input/output	
maximum input length	4 characters
maximum output length	4 characters
page size (in characters)	4 characters
terminal type (from user system	20 characters
NDL; left-justified)	

<optional text>

21VVCC (Connect Reply)

Sent from server system to user system to respond to a
Connect message. States whether or not a terminal-to-program
connection has been established for virtual station number
"VV" (user system's transmit address) via the response code
"CC": "00" for success, "01" for invalid address or system
full, "02" for no MCS reading on station, and "03" for
already connected.

22VV (Disconnect)

Sent from user system to server system to break an existing
terminal-to-program connection for virtual station address
"VV". There is no reply.

23 (EOF Confirmation)

Sent by a system receiving a file, after completing all
processing of it, to confirm receipt of the sender's end of
file. Also sent at connection-establishment time if no file
is being received, to synchronize the two systems after a
crash. (See FILE TRANSFER RECOVERY for more information.) On
receipt, this message is ignored if no PUT is in progress.

The following chart of system messages depicts the requirement
for implementation of these messages for both the host system and

the remote terminal:

| Message | Host System | | Terminal System | |
Code	Recv	Send	Recv	Send
01	Not required	Mandatory	Mandatory	Not required
02	Mandatory	Mandatory	Mandatory	Optional
03	Not required	Mandatory	Mandatory	Not required
04	Mandatory	Mandatory	Mandatory	Mandatory
05-07	Mandatory	Optional	Mandatory	Optional
09	Mandatory	Optional	Mandatory	Optional
10-23	Optional	Optional	Optional	Optional

Note: Message codes greater than 03 are not implemented by
       DC1000. Message codes 10-19 are file-transfer related;
       if file-transfer is implemented, they are mandatory.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

FORMAT OF PUT/FETCH MESSAGES

Following is the format of the Put and Fetch messages, starting
with the first character after the system message code. The put
message is identical to the Fetch message except for the message
code (note that the Put message has no logical need to specify
the source file identification; this is included for
consistency). All numerical values are specified in character
representation of decimal values, and have maximum values of all
9's.

Charact (35 a)

```
01  PUT_OR_FETCH_RECORD,
  02 REQUESTOR              CHARACTER(6),
    03 REQUESTOR_TYPE       CHARACTER(1),
              % 0 = SPO user
              % 1 = user program
              % 2 = terminal user
    03 REQUESTOR_ID         CHARACTER(4),
              % program ident or terminal ident.
    03 REQUESTOR_DIRECTION  CHARACTER(1),
              % P = original request was "PUT"
              % F = original request was "FETCH"
  02 REPLACE_FLAG           CHARACTER(1),
              % 0 = do not replace existing file     "AS"
              % 1 = replace existing file (if any)   "ONTO"
  02 ORIGINAL_SYSTEM_TYPE   CHARACTER(4),
              % in the same format as
              % "CCCC" in station-id message
  02 BEGINNING_RECORD_NUMBER  CHARACTER(8),      Zero/1 relative
              % record number at which to
              % start transfer
  02 RECORD_COUNT           CHARACTER(8),
              % number of records in file
  02 RECOVERY_FETCH         CHARACTER(1),
              % 0 = not a recovery FETCH
              % 1 = this is a recovery FETCH      163
  02 SOURCE_FILE            CHARACTER(174),
    03 FILE_NAME            CHARACTER(100),
    03 DEVICE_KIND          CHARACTER(2),
              % 00 = disk
              % 01 = tape (any available)
              % 02 = 9-track tape
              % 03 = 7-track tape
              % 04 = phase-encoded tape
    03 RECORD_SIZE          CHARACTER(4),
    03 BLOCK_SIZE           CHARACTER(4),
    03 PACK_NAME            CHARACTER(17),
    03 FILE_KIND            CHARACTER(2),
              % 00 = code file
              % 01 = data file
              % (other types may be added)
```

```
       03 USERCODE          CHARACTER(17),
       03 PASSWORD          CHARACTER(17),
    02 DESTINATION_FILE     CHARACTER(174);
              % (format same as "SOURCE_FILE")
```

In this record, the source FILE_NAME is in whatever syntax is
appropriate for the source system and the destination FILE_NAME
syntax is appropriate for the destination system.   If an
implementer wishes to allow transfer of a file family, he must do
so by initiating one transfer for each member of the family. The
destination file must be created with the specified FILE_KIND, so
it will be of the same type as the source file.   Note that the
source and destination RECORD_SIZEs must be the same.   Note also
that a system is not required to implement all values for all
options; if it receives a request specifying a device kind, file
kind, or size parameter that cannot be handled, it may return an
error of "unimplemented option".

FILE TRANSFER

File transfer is supported for disk and tape devices, and files
may be transferred between any combination of these.   Tape
transfers are restricted in that only one file per tape may be
transferred, and library-maintenance (load-dump) tapes are not
supported (these tapes may be transferred by loading to disk
before transferring). Between dissimilar systems only code files
and 8-bit data files (including text files) are allowed. Similar
systems are permitted to agree to transfer other files (e.g.,
printer backup files, arbitrary frame-size files). The protocol
allows transfer of only one file at a time in each direction.

The user initiates a file transfer by entering a variation of the
CSG standard "COPY" statement. This command specifies source and
destination hosts for the transfer, by naming their "host names".
These two names must be the names specified in the Station-id
system messages at initialization.   If the specified source host
is the local host, the transfer is a PUT;   if the source is the
remote host, the transfer is a FETCH.

The actual transfer of a file is initiated by exchange of control
information between systems.   All transfers occur as PUTs;   if
one system wishes to do a FETCH, it sends a Fetch message to the
other system, which causes it to initiate a PUT sequence, sending
a Put message.   The Fetch and Put messages completely specify
both source and destination file.   The Put message elicits a Put
reply, signifying that data transfer may begin.   The sender then
sends one or more data messages (device address 04) followed by
an end-of-file message.

On completion of file transfer, the system that received the file
informs the original requestor of completion of the request.   If
the requestor is foreign to this (the receiving) system,   the
notification is by a User-destined message.

The exchange of messages for a file transfer is diagrammed below:

                        0011 Fetch, attributes

        0010 Put, attributes

                        0017 Put reply, success/failure code

     04 Beginning of header FSd

        04 File header data

        04 File header data

                    . . .

     04 Beginning of file FS 1

                04 Data

                04 Data

                    . . .

                    000504 Device not ready, suspend (if
                                                       needed
                    000704 Device ready   for flow
                                                       control)

                    . . .

                04 Data

        04 End of file FS 2

                    0023 EOF confirmation
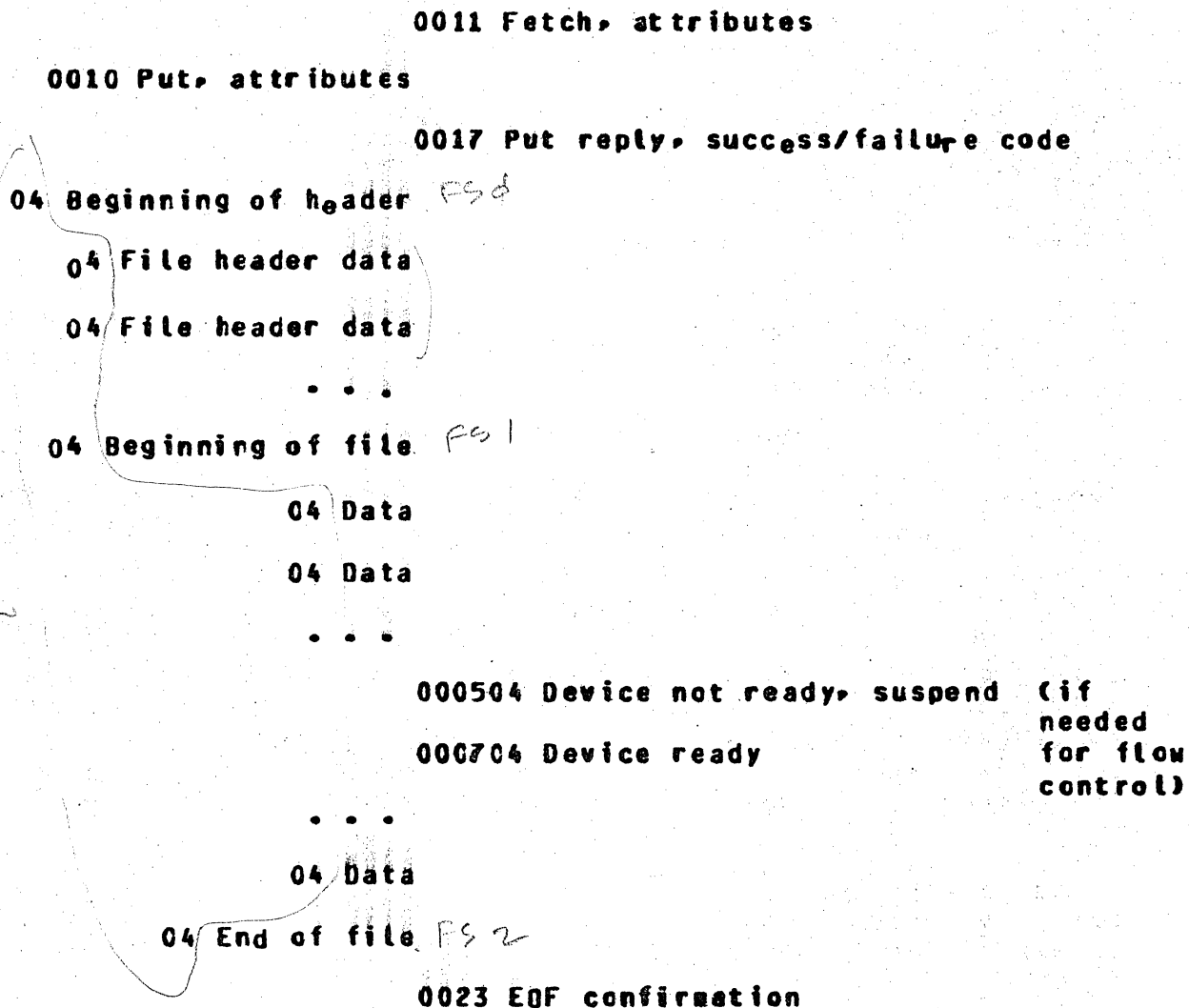
As shown in the diagram, the 0005 "Device not ready - suspend"
and 0007 "Device ready" messages are used for flow control.

MAINTENANCE OF FILE CHARACTERISTICS

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

Specifically, if a file is transferred between two like systems,
even if via one or more unlike systems, all of its
characteristics must be preserved. To accomplish this, a

symbolic version of the file's file header (understandable to all systems) is transferred with the file. This symbolic file header is the same as that used by the CSG standard library tape format. It is prefixed in the data stream by a beginning-of-header FS 0 message and may comprise one or more data messages.

When the symbolic file header is received, it is parsed and as many of the file characteristics therein as make sense on the receiving system are incorporated into the file being created (this may be all the characteristics or none). If the file is transferred to a system type other than the one where it was originally created, the symbolic file header and 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, so as to look identical to the original file, with the exception of filename (this is conveyed in the PUT_OR_FETCH_RECORD).

Not every file transfer will include a symbolic file header. Some systems may not be able to save symbolic headers, and so will not be able to forward them on subsequent transfers (but this must be advertized). Also, a foreign-created file may be modified or replaced locally, thus invalidating the saved symbolic header. In this case, the symbolic header may be omitted, or used to describe the file according to its new creation system type.

The means of saving the symbolic file header and associating it with the file are implementation choices, but here are four suggestions, the first three of which probably require MCP changes.

   1. Save the symbolic header as an extension of the real header, maintained by the MCP.

   2. Save the symbolic header within the data of the file, but hidden from all programs except SYCOM (for example, if the file is FORTRAN source, the symbolic header must be invisible to the FORTRAN compiler).

   3. Save the symbolic header in a secondary file given a unique name that can be inferred from the name of the data file. In the case of tape volumes, the secondary file could be the second file on the tape, after the data file.

   4. Save symbolic headers for all files received by SYCOM in a common directory file maintained by SYCOM.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

Alternately, one such directory file could be maintained
on each volume, containing symbolic headers for data
files on that volume.

FILE TRANSFER RECOVERY

Any file transfer operations that cannot be executed when
requested because another transfer is in progress must be queued
(in a non-volatile medium) and executed when possible. (All
requests are queued as PUTs, so if a user requests a FETCH, the
request is sent to the other SYCOM, where it is queued as a PUT
on that system.)

During a file transfer, both SYCOMs must maintain enough
information that if one or both systems fail, the transfer can be
restarted at the point it was interrupted (rather than at the
beginning). This means that while sending a file, SYCOM must
remember (in a non-volatile way) that it is sending a file, and
that while receiving a file, SYCOM must remember (in a
non-volatile way) that it is receiving a file, and how many
records have been actually written to the destination file (the
non-volatile copy of this value need not always be absolutely
current).

Whenever the SYCOM connection becomes established (regardless of
which SYCOM came up first), each SYCOM looks to see it it was in
the process of receiving and/or sending a file. If in the
process of receiving a file, SYCOM sends a FETCH request
specifiying that this is a recovery request and specifying the
number of the record with which to restart the transfer. If not
in the process of receiving a file, SYCOM conveys this fact via
an EOF confirmation message to the other SYCOM for
synchronization purposes. If SYCOM sees at establishment time
that it was in the process of sending a file, it holds that
request in abeyance, pending receipt of the recovery FETCH
request. On receipt, it restarts the transfer at the specified
record number (it is recommended that this be done in a
direct-access way rather than skipping through the records to
reach the desired point, but whether and how this is done is left
to the implementor). (Note that since a given SYCOM can be
sending and receiving a file at the same time, it must make both
of the above checks at connection-establishment time.)

When SYCOM notices the other system breaking the connection on a
switched line, it should wait for re-establishment and verify the
host name specified in the Station-id message. If connection is
re-established on the same line, but the host name is different,
all file transfer requests involving that host are purged.

TERMINAL SYSTEM MESSAGES

All instructions to or replies from the terminal are called local
messages. All local messages start with a period (.). Messages
not starting with a period are assumed to be input or output to
or from the host system. The following set of local messages is
recommended.

Note: These messages are meant to be a guideline and depend
         upon the particular needs of an installation.

    1.   The unit device identification scheme normally used by a
         terminal should be substituted in the following messages
         wherever "x" appears.

    2.   If device mnemonics are used, the following standard
         applies:

         a. CP - Card punch

         b. CR - Card reader

         c. LP - Line printer

         d. MT - Magnetic tape

         e. CS - Cassette

INPUT MESSAGES

.ABORT FETCH <operand>

                         Aborts the FETCH in progress, if any.
                         Initiates system message code 19.

.ABORT PUT <operand>

                         Aborts the PUT in progress, if any.
                         Initiates system message code 18.

Both .ABORT PUT and .ABORT FETCH include an optional operand
defined as follows:

<operand> := <number>   Will remove PUT or FETCH <number> from
                         the waiting schedule of files to be
                         transmitted.

<operand> :=   =   Will clear PUT or FETCH schedule.

<operand> := <filename> Will remove <filename> from the waiting
                         schedule of files to be transmitted.

BURROUGHS CORPORATION   COMPANY CONFIDENTIAL
COMPUTER SYSTEMS GROUP   B1800/B1700 SYCOM
SANTA BARBARA PLANT   SDS 2222 2590 (02)

<operand> :=   No operand indicates an ABGRT PUT or an
                          ABORT FETCH is in progress.

.BYE   Upon receipt of this command, SYCOM should
                          return control of this terminal to any
                          previous MCS.   SYCOM would be expected to do
                          an implicit .DISCONNECT if this terminal were
                          currently connected to another host.

.CL MT x   Causes the magnetic tape buffers associated
                          with drive x to be discarded and the tape
                          rewound.

.CL x   Command to be given to clear buffers in the
                          event device x becomes inoperable during
                          transmission or reception of data.

.CONNECT   *(this is out of list)*   Means connect the terminal or SPO as a
                          virtual terminal to the remote system (using
                          the virtual LSN concept).

.COPY filename [(USER=a,PASSWORD=b)]   [BINARY]

     AS
[filename [(USER=c,PASSWORD=d)]]
     ONTO

                                    PACK
                                    TAPE
      FROM [volume] ([KIND=   PETAPE   ,] HOST=hostname)
                                    TAPE7
                                    TAPE9

                                    PACK
                                    TAPE
        TO [volume] ([KIND=   PETAPE   ,] HOST=hostname)
                                    TAPE7
                                    TAPE9

     where:
        {}   denotes choice of items.
        []   denotes optional item.
        name   denotes variable name.
        NAME   denotes literal text.
        "AS"   means "create a new file of this name".
        "ONTO"   means "write over existing file of this name".
        "volume"   defaults to system disk (if "KIND=" is also
                          defaulted).
        "KIND="   defaults to "KIND=PACK".
        "DISK"   is an abbreviation for "KIND=PACK".
        "USER=a/b"   is an abbreviation for "USER=a,PASSWORD=b".
        "filename/="   is to be interpreted as the whole directory
                          ("=/=" only to be allowed if a usercode is
                          supplied).

not yet done

Transfers the specified file in the direction
specified by the source and destination
hostnames. User/password values identify the
user for purposes of verifying access rights
to the specified file, and perhaps also
affect defaults for parts of the filename.
Source user/password default to those of the
logged-in user, and destination user/password
default to those of the source. Destination
filename defaults to that of the source (in
which case the "AS"/"ONTO" option defaults to
"AS"). Initiates system message code 10 or
11.

.DISCONNECT   This is the inverse of .CONNECT and
                               disconnects the virtual terminal from the
                               remote system.

.EOJ   This entry will cause a terminal to revert to
                               a non-SYCOM mode if appropriate, and sends a
                               CLOSE message to the remote system to request
                               it do the same. This entails an orderly
                               cycling down of jobs and terminals before
                               coming to END-OF-JOB.

.IGNORE x   This message can be entered in response to
                               the local output message:

                                    .MT x PE   Where: x = Unit #

                               It causes the last record to be re-read
                               ignoring any parity errors.

.LIST PARAMETERS
     PUT <operand>
     FETCH <operand>

                               See definitions of <operand> from .ABORT.
                               This command will display the local system
                               PARAMETERS at either a terminal or a SPO. If
                               .LIST PUT or .LIST FETCH is entered from a
                               terminal, the waiting schedule of a user's
                               file transfer request as specified by the
                               <operand> will be displayed. If entered from
                               the SPO, the waiting schedule of all file
                               transfer requests as specified by the operand
                               will be displayed.

.OFFLINE   This is the inverse of .ONLINE and breaks the
                               connection if access was via a switched line.
                               The program remains in the mix.

.ONLINE [hostname] Places the local system in the online mode,

attempting to establish or re-establish communications with the specified host system. If no hostname if specified, or the last-named host is specified, re-establishes connecton to the last-named host (if connection was broken for some reason). If a hostname other than the current one is specified, this command breaks connecton to the current host if necessary, and establishes connection to the specified host. (If the host system is accessed via a switched line and its telephone number has been specified via .PHONE, .ONLINE initiates dial-out.) Invalid occurrences of this command are: the first .ONLINE has no hostname, an .ONLINE command with no hostname when the connection already exists, and an .ONLINE command specifying the current host when the connection to it already exists.

.PHONE
.PHONE =
.PHONE [new phone number]

If an auto-dial unit is supplied, specifies the telephone number to be used the next time dial-out is initiated to establish connection to the specified host. (Dial-out is initiated by the .ONLINE message.) The .PHONE with no operand will display the current phone number. .PHONE = will remove and clear the current phone number and .PHONE <new phone number> will enter into the system the new phone number, replacing any previous number. An optional "=" may be inserted between .PHONE and <new phone number>.

.PURGE x tttttt

Write a tape label on magnetic tape drive x with the ID of tttttt. The following rules apply:

1. The tape label (tttttt) must be six characters. If less than six characters, the tape is considered a scratch tape.

2. If the tape is in use when the command is issued, the MCP will ignore the command.

3. If no label (tttttt) is specified, the tape is made into a scratch tape.

.READ [CARD] <filename>
    [DISK]
    [TAPE]
                This command initiates the transfer of a card
                deck to be run as a job from the card reader,
                a disk file or a tape file. If no device
                type is specified, "CARD" is assumed.
                "filename" is the fully-qualified file name,
                and if not specified, defaults to
                "RJE/CARDS".

.READY x   Read tape label on magnetic tape drive x and
                display label on SPO.

.RETRY x   Informs the system to retry reading the
                record in question from magnetic tape drive
                x. This entry can be made in response to
                local output message .MT x PE.

.RY x   Informs the SYCOM terminal system that unit x
                is available and ready. Initiates system
                message code 07.

.SIZE=nnn   Informs the SYCOM terminal system to set the
                block size of information to be transmitted
                to the central system, equal to nnn. The nnn
                must be in the range of 80 to 402, inclusive.
                Initiates system message code 02.

.STOP   Terminate program immediately.

.SV x   Directs the terminal system to inform the
                host system to cease transmitting or
                receiving data from device x and to save the
                entire current file. Initiates system
                message code 06.

The .ABORT FETCH, .ABORT PUT, .BYE, .CONNECT, .COPY, .DISCONNECT,
.LIST and .PHONE commands should be the only ones allowed from a
terminal. All commands are allowed from a SPO.

OUTPUT MESSAGES

Under certain conditions, the SYCOM terminal system will supply
information to the operator via the console typewriter. Some of
these messages require an action by the operator. These messages
will be identified in the operator's manual. For each of the
SYCOM output message listed, a code number is given. If the
message is returned to a program rather than displayed on a
console, this code number will be included to avoid the need for
the program to parse the text of the message.

00   .CP-CHK   The card punch has detected a punch
                          error on the last card punched.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (C2)

01   .CP-NR   The card punch is in a not-ready state.

02   .CR-CHK   The card reader has detected a read
                         error on the last card read.

03   .CR-INV   The card reader has detected an invalid
                         punch in a column other than column 1.

04   .CR-NR   The card reader is in a not-ready state.

05   <ERR>   The error pushbutton was depressed.
                         Re-enter all information.

06   .HOST EST   Informs the operator that the system has
                         established   or   re-established
                         communications with the host system.

07   .HOST TERM   The terminal system has been placed in
                         an off-line mode due to:

                         1.   Having   received   an   End   of
                              Transmission indication from   the
                              host system.

                         2.   Direct   result   of   keyboard   entry
                              DISCONNECT, which causes termination
                              of all activity.

08   .INV-KBD   Indicates   that an invalid operation has
                         been performed on keyboard input,   such
                         as:

                         1.   Attempting   to   type in more than 80
                              characters for an input message.

                         2.   Enabling keyboard input   and   typing
                              at a rate of less than one character
                              every ten seconds, or not depressing
                              the EOM button   at   completion of
                              input.

                         3.   Input   of   a   message   which   is
                              inappropriate,   e.g.,   typing
                              ".ONLINE" when the system is already
                              online to the specified host.

                         4.   Input of a message which is not
                              defined.

                         5.   Incorrectly inputting a message.

                         6.   Attempting to type in a message for
                              the   host   system   when   sufficient

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

space is unavailable.

7. An error condition occurred during keyboard input, or an invalid key was depressed. Re-enter all information.

09 .LP-CHK — The line printer has detected a print error on the last line printed.

10 .LP-NR — The line printer is in a not-ready state.

11 .MT x BE — Number of blocks that have been read is incorrect.

12 .MT x FE — Magnetic tape has incorrect format.

1. Record size not correct, or

2. No ending label or file mark.

13 .MT x FT — Foreign magnetic tape. Record format incorrect.

14 .MT x NL — No label on tape or magnetic tape drive x.

15 .MT x NR — Magnetic tape drive x not ready.

16 .MT x PE — Tape parity error on magnetic tape drive x.

17 .MT x REEL — Another reel of tape is required on magnetic tape drive x.

18 .MT x WI — Tape on unit x is write inhibited.

19 .MT x zzzzzz/mmmmm #rrr (ddddd) Tape label information pertaining to tape on magnetic tape drive x:

zzzzzz = File name

mmmmm = Multifile id or zeros

rrr = Reel number

ddddd = Creation date of SYCOM system

The following messages are self-explanatory:

20 FILE <filename> NOT FOUND

21   FILE <filename> LOCKED

22   FILE <filename> FILE TYPE CANNOT BE TRANSMITTED

23   CAN'T SCHEDULE NOW; TRY AGAIN LATER

24   REQUEST SCHEDULED; WILL SEND FILE <filename> ASAP

25   FILE <filename> START TRANSMISSION

26   FILE <filename> END TRANSMISSION

27   ONTO-PACK OR USER-CODE NOT FOUND--COPY ABORTED

28   FILE <filename> IN USE AT DESTINATION--COPY ABORTED

29   FILE <filename> RECORD SIZE IS TOO BIG

30   FILE <filename> TOO MANY RECORDS

31   REQUEST SCHEDULED--REMOTE RJE OFFLINE

32   UNABLE TO ESTABLISH COMMUNICATION

33   UNABLE TO DIAL OUT

34   NOW DIAL HOST

35   ALREADY CONNECTED; CAN'T DIAL

36   DIAL REQUESTED BUT NO ACU DEFINED

37   INVALID PHONE NUMBER

38   SYCOM IS CLOSING; CANNOT ACCEPT INPUT

39   General syntax error (exact text explains further)

40   SYCOM CLOSING ON REQUEST FROM REMOTE SYSTEM

41   SYCOM WILL CLOSE WHEN NEXT IDLE

42   LOST CONNECTION TO REMOTE SYSTEM SYCOM

43   REMOTE SYCOM HAS GONE DOWN AND COME BACK UP

44   THERE IS NO 'PUT' IN PROGRESS TO ABORT

45   PUT OF FILE <filename> ABORTED

46   THERE IS NO 'FETCH' IS PROGRESS TO ABORT

47   FETCH OF FILE <filename> ABORTED

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

48   SYCOM NOW ONLINE TO HOST <hostname>

COMMUNICATION ERROR MESSAGES

Error reporting may be handled one of two ways.   The following
errors may be reported individually as they occur:

49   .DSET-NR   Data Set (Modem) is in a not-ready
                            state.

50   .LN-ERR   Buffer currently transmitted by the
                            terminal system has been discarded
                            because retry count has been exceeded.

51   .NSP-EXP   In the process of transmission of data
                            to the host, the system has exceeded the
                            allowable number of NO RESPONSE retries.

52   .RX-RTYS-UP   In the process of data reception from
                            the host,   the system has exceeded the
                            allowable number of error retries.

53   .TX-RTYS-UP   In the process of transmission of data
                            to the host, the system has exceeded the
                            allowable number of retries.

Alternately, all errors may be reported by the following message:

54   .RETRYS UP

HOST SYSTEM MESSAGES

Applicable host system messages are found in the appropriate host
master control program or language operations manuals.

ACCEPTABLE CHARACTERS FOR TERMINAL SPO

The following list constitutes the set of characters which are
acceptable for transmission from a remote station's SPO.   The
entry of any character not listed shall either be deleted from
the message prior to transmission, or shall invalidate the entire
message prior to transmission, or shall invalidate the entire
message at the remote station.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (C2)

```
0 through 9
A through Z
                    Blank or space
+                   Plus
-                   Minus or hyphen
*                   Asterisk
/                   Virgule (slash)
=                   Equal sign
$                   Dollar sign
,                   Comma
.                   Period
;                   Semicolon
:                   Colon
"                   Quote mark
(                   Left parenthesis
)                   Right parenthesis
>                   Greater symbol
<                   Less symbol
@                   At sign
                    Exclamation mark
[                   Left bracket
]                   Right bracket
'                   Apostrophe
?                   Question mark
#                   Number mark
&                   Ampersand
%                   Percent sign
\                   Reverse virgule (slash)
↑                   Up-arrow
                    Left arrow or shifted 0 (Reversed for
                    backspace function; nontransmittable)
    LF              Line feed (nonprinting)
    BELL            Control G (nonprinting)
```

The following is a table of equivalence for allowable control characters.

    Carriage Return/Control C/ETX/End of Message

    Rubout (Delete)/Control U/Error

    Left Arrow/Shifted 0/Backspace

NOT grave accent / tilde / left brace / right brace } for terminals

allow will do

SYCOM TRANSMISSION OF PACKED DATA

DATA RECORD FORMAT

As in previous version of this protocol, data records within the same transmission are separated by RS characters, and repeated blanks are compressed with the ESC-integer mechanism, but the method of representing non-printable characters is changed. The printable characters are those in the above list (acceptable characters for terminal SPO), except LF and BELL.

If the record contains only these characters, the first character of the record is not a DLE. If the record contains any character not in this list, the first character of the record is a DLE, 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. This mapping is accomplished by prefixing the 4 bits of data with 4 bits of hex 4.

TERMINAL-TO-PROGRAM COMMUNICATION

The protocol supports a user at a terminal in one host connecting to an interactive applications program in another host, and using it as if he were locally attached to the program. The main goal of this protocol is to minimize the visibility of the interfacing software, such that it appears to the user as if were directly connected to the applications program, and such that existing applications programs (e.g., CANDE) may be utilized with a minimum of modification.

Some ARPA Network terms that are useful in discussing this topic are "user" host and "server" host. The "user" host is the host with the users at terminals wishing to engage in interactive computing, and the "server" host is the host that supplies the interactive service.

The method for conveying terminal data between systems makes use of the device address field, DA1 DA2, to specify the terminal with which the data is associated. "DA1, DA2" in this context is termed a "virtual station address", and is the name used by user and server hosts to refer to a particular terminal. The addressing scheme is as follows:

1. DA1 and DA2 are ASCII letters, one upper case and the other lower case.

2. If "DA1, DA2" is the transmit address for a particular terminal, then "DA2, DA1" is the receive address for that terminal.

3. When the user host refers to a terminal (by transmit address), "DA1" is a lower-case letter and "DA2" is an upper-case letter. When the server host refers to the same terminal, since the roles of "transmit" and "receive" are reversed, "DA1" is an upper-case letter and "DA2" is a lower-case letter.

4. If a user host supports "N" different terminals connecting to a particular server host, the user host refers to them by the following virtual station addresses (as transmit addresses):

    aA, bA, cA, ..., zA, aB, bB, cB, ...,

    and the server host refers to the same terminals by these virtual stations addresses (also as transmit addresses):

    Aa, Ab, Ac, ..., Az, Ba, Bb, Bc, ...

The user and server host decide independently how many terminals

they will support, so these numbers will likely be different.

The sequence of events in a remote interactive session are as follows:

connection: when SYCOM establishes connection to a host in response to an .ONLINE command, it also attempts to establish a terminal-to-program connection, by assigning a virtual station address to this particular user and specifying it in a Connect message to the server host. This Connect message also describes the terminal's characteristics.

The server host receives the Connect message and checks to see if it can support the new connection; returns a Connect Reply message specifying whether or not it can. If the connecton is successful, the server host makes this virtual station appear just like a real hardware terminal on that system that has been powered up and recognized as ready (but not yet "signed on"). It makes the station ready in a way determined by the nature of the host's operating system. The server host also saves the terminal characteristics specified in the Connect message, and makes them accessible to user programs, preferably by storing them in the NDL tables.

disconnection: whenever SYCOM breaks connection due to a .OFFLINE, .EOJ, .STOP or .ONLINE command to a new host, it also breaks the terminal-to-program connection (if any) by sending a Disconnect message to the server host, specifying the virtual station address.

The server host receives the Disconnect message, and does whatever is appropriate in that system to make that virtual terminal appear as though it was powered off, or otherwise available for re-use.

during the life of the connection: if the terminal-to-program connection was established successfully, all keyboard input entered by the user at his terminal (except for lines beginning with a period, which are interpreted locally) is transmitted to the server host, specifying the virtual terminal address as an RJE device address. The server host presents this input to whatever applications program or MCS that is currently attached to this terminal.

Any terminal output text that is generated by the MCS or applications program is transmitted back to the user host, specifying the virtual terminal address as an RJE device address. The user host SYCOM then displays this data on the user's terminal.

ONE POSSIBLE IMPLEMENTATION

The following is an outline of one possible method of implementing this protocol. It is presented here for

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 SYCOM
SDS 2222 2590 (02)

illustrative purposes only; it is not the responsibility of this
document or the RJE committee to develop program logic for the
individual plants.

The user host decides how many outbound connections will be
allowed, and in the NDL station section defines one station for
each, specifying a receive-address/transmit-address pair of
"Aa"/aA" for the first, "Ab"/"bA" for the second, and so on.
Because of where these declarations fall within the station
section, they are assigned some logical station numbers. In the
file section, a file family is declared to have these logical
stations as members.

At initialization, SYCOM (or some other MCS written for this
purpose) opens the remote file pointing to these stations, and
does a "remote file info" operation to obtain from NDL the LSNs
for these virtual stations; it places these numbers in a table
to map the LSN of each user's terminal into an LSN to reference
the corresponding virtual station.

When a user wishes to connect to the remote host, SYCOM finds the
next available entry in his LSN table, and issues a Connect
message for that virtual station address. On receipt of the
Connect Reply, SYCOM reports the outcome to the user.

During the life of the connection, SYCOM transmits unexamined any
data entered by the user at his terminal (except commands
prefixed by the SYCOM signal character) to the server, by doing a
remote write to the LSN of the user's virtual station. In
processing this remote write, NDL maps the LSN into the
appropriate virtual station address and transmits the data
according to the SYCOM protocol, with the virtual station address
as the SYCOM device address. Any data read by the user NDL from
the SYCOM line with a device address like "Aa", "Ab", etc., is
presented to SYCOM, via his remote file, as input from the
appropriate LSN for this virtual station. SYCOM then
interrogates the LSN table to find the real LSN of the associated
user, and writes the data to him via a remote write.

When the user wishes to disconnect from the remote host, SYCOM
sends a Disconnect message to the server, specifying the virtual
station number of this user.

The server host decides how many inbound connections will be
allowed, and in the NDL station section, defines one station for
each, specifying a receive-address/transmit-address pair of
"aA"/"Aa" for the first, and so on. Any program wishing to serve
these stations (e.g., SMCS) will include them in the family for
their remote file declaration.

The server NDL intercepts Connect, Disconnect, Device Not Ready -
Suspend, and Device Ready messages (unlike all other device
address "00" messages, which are routed to SYCOM). On receipt of
a Connect message, NDL verifies that the addressed virtual

station exists and that some program is ready to read from it. A
Connect Reply message is sent indicating success or failure of
the Connect.

During the life of the connection, any input data received by the
server NDL for an enabled virtual station is conveyed to the
waiting applications program through his remote file in exactly
the same manner as for local terminals.   Any output data
generated by the program and written to the virtual station
through his remote file is transmitted to the user host, with the
virtual station address as the RJE device address.

PROGRAM-TO-PROGRAM COMMUNICATION

A limited type of program-to-program communication is possible by
adapting the above scheme for terminal-to-program communication.
The RJE MCS will provide a mechanism (of implementation-dependent
nature) to allow a local program to masquerade as a terminal.
This user program will format messages as if they were keyboard
input, and the RJE system will send them to the server program.
Likewise, the server program's output will be presented to the
user program as if it were terminal display data.

APPENDIX A - 1967 ASCII AND EBCDIC CHARACTER ASSIGNMENTS

1967 ASCII	EBCDIC	EBCDIC GRAPHIC	EBCDIC PUNCH CODE
00	00	NUL	12-0-9-8-1
01	01	SOH	12-9-1
02	02	STX	12-9-2
03	03	ETX	12-9-3
04	37	EOT	9-7
05	2D	ENQ	0-9-8-5
06	2E	ACK	0-9-8-6
07	2F	BEL	0-9-8-7
08	16	BS	11-9-6
09	05	HT	12-9-5
0A	25	LF	0-9-5
0B	08	VT	12-9-8-3
0C	0C	FF	12-9-8-4
0D	0D	CR	12-9-8-5
0E	0E	SO	12-9-8-6
0F	0F	SI	12-9-8-7
10	10	DLE	12-11-9-8-1
11	11	DC1	11-9-1
12	12	DC2	11-9-2
13	13	DC3	11-9-3
14	3C	DC4	19-8-4
15	3D	NAK	9-8-5
16	32	SYN	9-2
17	26	ETB	0-9-6
18	18	CAN	11-9-8
19	19	EM	11-9-8-1
1A	3F	SUB	9-8-7
1B	27	ESC	0-9-7
1C	1C	FS	11-9-8-4
1D	1D	GS	11-9-8-5
1E	1E	RS	11-9-8-6
1F	1F	US	11-9-8-7
20	40	SP	No Punches
21	4F	Exclamation Point	12-8-7
22	7F	"	8-7
23	7B	#	8-3
24	5B	$	11-8-3
25	6C	%	0-8-4
26	50	&	12
27	7D	'	8-5
28	4D	(	12-8-5
29	5D	)	11-8-5

1967 ASCII	EBCDIC	EBCDIC GRAPHIC	EBCDIC PUNCH CODE
2A	5C	*	11-8-4
2B	4E	+	12-8-6
2C	6B	,	0-8-3
2D	60	-	11
2E	4B	.	12-8-3
2F	61	/	0-1
30	F0	0	0
31	F1	1	1
32	F2	2	2
33	F3	3	3
34	F4	4	4
35	F5	5	5
36	F6	6	6
37	F7	7	7
38	F8	8	8
39	F9	9	9
3A	7A	:	8-2
3B	5E	;	11-8-6
3C	4C	<	12-8-4
3D	7E	=	8-6
3E	6E	>	0-8-6
3F	6F	?	0-8-7
40	7C	@	8-4
41	C1	A	12-1
42	C2	B	12-2
43	C3	C	12-3
44	C4	D	12-4
45	C5	E	12-5
46	C6	F	12-6
47	C7	G	12-7
48	C8	H	12-8
49	C9	I	12-9
4A	D1	J	11-1
4B	D2	K	11-2
4C	D3	L	11-3
4D	D4	M	11-4
4E	D5	N	11-5
4F	D6	O	11-6
50	D7	P	11-7
51	D8	Q	11-8
52	D9	R	11-9
53	E2	S	0-2
54	E3	T	0-3
55	E4	U	0-4

1967 ASCII	EBCDIC	EBCDIC GRAPHIC	EBCDIC PUNCH CODE
56	E5	V	0-5
57	E6	W	0-6
58	E7	X	0-7
59	E8	Y	0-8
5A	E9	Z	0-9
5B	4A	[	12-8-2
5C	E0	\	0-8-2
5D	5A	]	11-8-2
5E	5F	up arrow	11-8-7
5F	6D	underline	0-8-5
60	79	back apostrophe	8-1
61	81	a	12-0-1
62	82	b	12-0-2
63	83	c	12-0-3
64	84	d	12-0-4
65	85	e	12-0-5
66	86	f	12-0-6
67	87	g	12-0-7
68	88	h	12-0-8
69	89	i	12-0-9
6A	91	j	12-11-1
6B	92	k	12-11-2
6C	93	l	12-11-3
6D	94	m	12-11-4
6E	95	n	12-11-5
6F	96	o	12-11-6
70	97	p	12-11-7
71	98	q	12-11-8
72	99	r	12-11-9
73	A2	s	11-0-2
74	A3	t	11-0-3
75	A4	u	11-0-4
76	A5	v	11-0-5
77	A6	w	11-0-6
78	A7	x	11-0-7
79	A8	y	11-0-8
7A	A9	z	11-0-9
7B	C0	left brace	12-0
7C	6A	lozenge	12-11
7D	D0	right brace	11-0
7E	A1	tilde	11-0-1
7F	07	DEL	12-9-7

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

INDEX