15 pp

2233 2886

Burroughs Corporation **B**

COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COBOL74 MCS

PRODUCT SPECIFICATION

REV LTR	REVISION ISSUE DATE	APPROVED BY	REVISIONS
A	11/2/81	*R.Stolt*	Original Issue -- Mark 11.0 Release

SBP 1060 8-76

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

TABLE OF CONTENTS

INTRODUCTION

GENERAL

The COBOL 74 MCS (hereafter just MCS) was implemented to demon-
strate all data communication facilities via COBOL 74. It
provides many features, the most important of which is that the
source code will be released as well as the object. This will
allow the user to see examples of the various MCS functions used
to support COBOL 74.

The MCS was implemented to be run under the control of SMCS. To
use it without the aid of SMCS would only inconvenience the user.
It is important that there be only one copy of the MCS running at
any one time.

OPERATING INSTRUCTIONS

The MCS was designed to run along side SMCS and thereby takes advantage of its presence. To use the MCS for the first time, you first have to make an entry in SMCS's "JOBS" file. (This should include 'COPIES=1' when running with the 11.0 release of SMCS, and 'LOG-ON' all the time.) For example:

     $ MCS74 LOG-ON COPIES=1 ; EXECUTE MCS74 PR 12 ;

Then one would logon to SMCS, and enter 'ON <MCS designator>'. The MCS will then be executed and open a remote file that will be attached to your station.

Suppose we had to run an inventory program called "INVENT", which requires INITIAL INPUT. The MCS requires that all such programs have an entry in a file called MYNAME/"INITINPUT". (See the section on INITIAL INPUT FILE FORMATs.) The user would make an entry in the MYNAME/"INITINPUT" file with the chosen designator of "I" to execute the program "INVENT". For example:

     # I ; EXECUTE INVENT ;

At the remote station, one would specify "# I ", which indicates that the following message is for the program associated with "I". Then the user would decide which queue that the program expects to read the message out of, and that is specified by combinations of "P"s and "S"s. These letters indicate the choices between primary and secondary queues. Four levels are possible, but those that are left off the right end are filled with "P"s. That also implies that if the queue specifier is left out entirely, "PPPP" is assumed. Following this is the message. The message can be up to the balance of the screen. Ensuing messages that are to be delivered to the same queue and the same program need not have those specified. The primary and secondary queue names are defines called "PRIMARYQNAME" and "SECONDARYQ-NAME".

They are currently defined as "HIGH" and "LOW" respectively.
That implies that "PPSP", for example, designates the queue name
"HIGH"/"HIGH"/"LOW"/"HIGH" and "PS" designates the queue name
"HIGH"/"LOW"/"HIGH"/"HIGH". Here is the form:


```
 # --- <progam designator> -------------------------- <message> ------|
                          |                           |
                          |- <queue designator> -|
```

Each is separated by one or more blanks. For example:

    # I PPSS This is the initial input message.

This specification will cause the message "PPSS This is the
initial input message." to be enqueued in the queue named "HIGH"/
"HIGH"/"LOW"/"LOW".

The MCS has six files declared. Externally their names are
MYNAME/"RMT", MYNAME/"TRACE", MYNAME/"SCHQ", MYNAME/"JOBQ",
MYNAME/"OVFLQ", MYNAME/"INITINPUT", and MYNAME/"SEGFILE". MYNAME
is a string of characters which CAN be the MCS' name. (Its
default is "MCS74". MYNAME/"TRACEFILE" is a printer backup file
that is created upon demand by one of two different ways; the
specification of the "TRACE" command or the setting of program
switch 0 equal to 1 prior to the execution of the MCS.
MYNAME/"SCHQ" and MYNAME/"JOBQ" are queues that are used by the
MCP to return job spawning information. MYNAME/"INITINPUT" is
the name of the file used to specify which job should be spawned
due to an initial input (see format). MYNAME/"SEGFILE" is the
name of a diskfile that is used to temporarily store SEGMENTs of
messages up until the entire message has been received by the
MCS. MYNAME/ "OVFLQ" is to handle an overflow of messages from
MYNAME/"REMOTE".

Specification of a string to be spawned by the MCS is accomp-
lished when the token following the pound sign ("#") is a ques-
tion mark ("?"). The balance of the message will then be
spawned. For example:

    #? EXECUTE HS/(HS)/CM1010 ;

COMMANDS

There are several commands available, such as "PASSWORD", "BYE", "ASSIGN", "STOP", "TRACE", "ESI", "STATUS" and "HELP". All of the commands are delimited by blanks. They can be used from a station if preceded by a pound sign ("#"). Following is a description of each:

PASSWORD

Specification of "PASSWORD" only will return the current setting. If the command is followed by "-", or ten blanks, the password will be cleared. If the command is followed by a token consisting of up to ten characters (not including embedded blanks), the password is set.

```
PASSWORD --------------------------------------------------------------
         |                    |
         |--- <password> ---|
         |                    |
         |-------- - --------|
```

STOP

The STOP command is used to begin an immediate shutdown of the
MCS and all of the jobs still executing under it (that do not
have NODIF specified).

    STOP ---|

BURROUGHS CORPORATION   COMPANY CONFIDENTIAL
COMPUTER SYSTEMS GROUP   B1800/B1700 COBOL74 MCS
SANTA BARBARA PLANT   P.S. 2233 2886 (A)

TRACE

Specification of this command alone responds with its mode.
Following the command with ON or OFF respectively sets the mode
as desired.   Another means for starting the trace facility (at
BOJ of the MCS only) is setting program switch 0 equal to 1.

The printer backup file that is created has the following format:


```
                          [time]..character(        12)
                            filler............(        2)
              [sourceordestination]............(        10)
                            filler............(        1)
                 [traceline]............(LINESIZE)
                            [size]............(        2)
                            filler............(        5)
              ------------------------------
                                                132 characters
```

The [sourceordestination] field contains the name of the source
or destination followed by an arrow to the right or the left.
The possible categories are:"ODT", "SCHQ", "RMT", "JOBQ", "INIT",
"ZIP", "SEG", "OVFQ" and a special case "DEBUG". "DEBUG" is the
only category that is followed by asterisks instead of an arrow.
For example (wrapped around):


```
    03:04:12.1AM   RMT   in--> 0100170011007000000000??000000000
    00000000045??????# JUNK PPSP                               61
    03:04:23.3AM   INIT in--> # JOB1 ; EX JOB1;
                                        00000100               80
    03:04:28.0AM   DEBUG **** JOBINFILE = 1
```


The [traceline] field consists of one or more lines of LINESIZE
(default=100) byte segments of the actual data traced.   (This
also includes the 50 byte datacomm header.)

The [size] field shows the byte length of the [traceline] field.


```
    TRACE ----------------------------------------------------------------|
                |                 |
                |---- ON -----|
                |                 |
                |---- OFF ----|
```

BYE

The BYE command causes that station to be detached from the MCS.
It cannot be used from the ODT.   Specification of this   command
also UNASSIGNs the station and if no other stations are currently
on the MCS, it will go to eoj.

     BYE ---|

ASSIGN

The ASSIGN command is used to cause all the input from a partic-
ular station to be enqueued into the specified queue. The form
is 'ASSIGN <qname>'. If <qname> equals a hyphen ("-"), then the
station will be unassigned. Specification of the STATUS command
will display assignments. The <qname> can be of any form, and
any length up to 48 characters including embedded blanks. This
<qname> is used as given, and does not have to be one of the
eight primary or secondary queues.

```
   ASSIGN  ----------------------------------------------------------------|
              |                        |
              |----  <qname>  -------|
              |                        |
              |-------- - ----------|
```

ESI

Specification of ESI alone will return the value of the End-of-
Segment-Indicator character. The default value is ETX (@03@).
The display will be as a character, (some characters are filtered
out by SYSTEM/ODT before they are displayed). If a character
follows the command the ESI value will be changed to that char-
acter. If a hexadecimal string ("@<two graphics 0-9 and A-F>@")
is specified because the desired ESI value does not have a
graphic, then the ESI value is set to that value. For example:

        ESI @40@

will set the ESI character to the value of blank. The EMI and
EGI values are also set to a default of @03@. These can only be
changed by changing the source records that certain to their
definition, and re-compiling.

    ESI --|
            | |
            |---- @ <two graphics 0-9 and A-F> @ ----|
            | |
            |---- < one character ESI > -------------|

STATUS

Specifiation of STATUS alone will return the status (ENABLED or DISABLED) and their ASSIGNments of all "known" lsns, while also specifying an lsn number, will only return the status of that lsn.


```
    STATUS ------------------------------------------------------------|
              |                           |
              |---- <lsn number > ----|
```

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 COBOL74 MCS
P.S. 2233 2886 (A)

HELP

Specification of HELP alone will return a list of legal commands,
while specifying a particular command will return the allowable
syntax and response for that command.

```
HELP ----------------------------------------------------------------|
           |                        |
           |--- HELP --------|
           |                        |
           |--- PASSWORD ---|
           |                        |
           |--- STOP --------|
           |                        |
           |--- TRACE ------|
           |                        |
           |--- ESI ---------|
           |                        |
           |--- BYE --------|
           |                        |
           |--- ASSIGN -----|
           |                        |
           |--- STATUS -----|
```

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B1800/B1700 COBOL74 MCS
P.S. 2233 2886 (A)

INITIAL INPUT FILE FORMAT

The format of the "INITINPUT" is very similar to that of SMCS's "JOBS" file. The format is as follows:

A pound sign ("#") must be in column one of every new entry. The next token must be the desirable string of characters that one wishes to use to specify initial input to the program that will be executed in that entry. Following that designator, a semicolon (";") is required. Following that can be all the necessary ODT commands, file equates, switch settings, etcetera to execute the desired job. This should end with another semicolon (";"). As many lines as is necessary can be used, but only the first need begin with a pound sign ("#"). The string, as it appears in the file will be spawned, so it is important to be sure that the syntax is correct.

INDEX