

PART 4

UPDATE PROCESSOR

The Update Processor is used to alter, by means of commands, existing sequential source or data files consisting of ASCII characters. These files may be of the following type and format:

- A file of type SC, with a maximum length of 80 characters per line;
- A file of type UF, with a maximum length of 132 characters per line.

If the lines are longer, they are truncated to the maximum length for the file.

Updating takes place at line level; however, character handling in the line is permitted in certain commands.

Commands may be given with keyword parameters or positional parameters. The keyword parameters may be given in any order. The positional parameters must be given in the order given in the description of the command. Mixing keyword parameters and positional parameters is allowed.

The input file, which may be on disc or on a non-disc device, is copied onto an output file, after the corrections given in the various commands are implemented. The output file may also be a disc or a non-disc device.

The functions available in the Update Processor are performed in two distinct phases:

- definition phase;
- execution phase.

Definition Phase

The definition phase of the Update Processor commences as soon as the Processor is activated.

Two commands, !!IN and !!OU, may be given to define the input file and the output file, where defining the input file must be the first command, if it is given, in the Update command stream.

The !!OU command may be given at any time in the definition phase. Only one !!IN and one !!OU command is allowed.

The input and output definition commands may be followed by the commands listed here. The parameters of these commands are placed in a definition phase table. Before any line of the original file is written to the output file, it is checked against the information in this table.

The definition phase commands are:

- !!IN Define the Input File
- !!OU Define the Output File
- !!RS Replace a Character String in all Lines
- !!DS Delete a Character String from all Lines
- !!DE Delete Lines Containing a Certain Character String
- !!IS Insert Lines After a Line Containing a Certain Character String.

Execution Phase

The definition phase is terminated as soon as one of the following commands is given.

Commands in the execution phase must be in line number order; if they are not, the command is rejected with an error message.

- !!RE Replace a Character String in a Line
- !!DL Delete one or more Lines
- !!IL Insert one or more Lines
- !!JN Join (Copy) Lines from an Auxiliary Input File
- !!EN Normal End of Updating
- !!EX Abnormal End of Updating
- !!KF Terminate Updating and Keep the Output File.

General Commands

These commands may be used at any time, during the definition or the execution phase:

- !!CC Change the Characters Identifying an Update Command
- !!CI Change the Command Input Device
- !!HL Help - Ask for the Syntax of Update Command(s)
- !!LF List the Input File.

The size of the Update Processor is 9 pages.

The Update Processor is called by the BCL command input from the device assigned to /EO or to /EE:

```
UPD [DUMP={ALL | PROG | NO}][,SIZE={MAX | <n>}]
```

The parameter DUMP indicates whether a dump must be made after an abort or an exit (LKM 3).

ALL Dump the monitor and the background machine in case of an abort, or if bit 8 is set in register A7 for LKM 3.
 PROG Dump only the background machine.
 NO No dump required. This is the Default.

The parameter SIZE reserves a work area of n pages of 2K words for Update. It is only useful when the parameter SIZE is not specified in the 'Declare Batch Processing Machine' command BCP.

MAX The system will reserve 32k words (16 pages) of work area for Update.
n A number, ranging from 0 through 16, specifying the number of additional pages required as work area. Default = 1 page.

If the UPD command is accepted and /EO is assigned to an interactive device such as console keyboard or display, the message:

UPD:

is output to that device. The user may now input one or more of the commands described in this chapter.

If the UPD command is rejected, then:

- If /EO is assigned to an interactive device, an error message explaining the error is output on this device, followed by the message:

UPD:

requesting the user to enter the correct UPD command.

- If /EO is assigned to a non-interactive device (such as a card reader), an error message is output on the ERR device, followed by the message:

UPD:

requesting the user to input the correct command from the device assigned to ERR.

- If /EO is assigned to a non-interactive device and there is no ERR device defined for this Job, the error message is output to the device assigned to /O2 and the Update processor exits. The BCP is reloaded and reads all subsequent commands on /EO until one of the commands :EOJ, :EOB or :STP is encountered.
- In some cases an erroneous input of command parameters may start the execution of the command. When the execution cannot be finished due to these parameters, the Update Processor will exit to the BCP. The user must then give a new command UPD.

Error Messages: See Appendix A.

Severity Codes

An UPD severity code is output when Update's processing is terminated. It is printed on the listing, and may be one of the following values:

- 0 Normal exit. The file is updated without processing errors.
- /60 Abnormal exit. No updated file is produced, as an error occurred during updating.
- /70 Abnormal exit. The Update Processor cannot be started.

DEFINITION COMMANDS

Definition commands must precede execution commands, as the definition command parameters are placed in a table. Prior to writing a line to the output file (i.e. after performing any execution command (except !!IL) relevant to that line), UPL will examine the definition command table to see if that line, and the lines preceding it but not yet written on the output file, need further modifications.

Definition commands are character string handling commands.

Character string parameters in the commands must be enclosed in inverted commas. If a string contains an inverted comma, this comma must be enclosed by two inverted commas.

Example: character string: l'hotel
 string parameter: 'l''hotel'

One of the error messages shown below is output if a definition phase command is rejected:

```
OVF. ON xy ALLOC            where xy may be RS, DS, IS or DE  
FROM VAL. NOT < TO VAL.
```

Syntax with keyword parameters:

```
type 1)  !!IN FNAME=fc
type 2)  !!IN FNAME=fn[,TYPE={SC | UF}][,USID=u][,DAD=d][,VERS=v]
```

Syntax with positional parameters:

```
type 1)  !!IN fc
type 2)  !!IN fn[,{SC | UF}][,u][,d][,v]
```

Use:

This optional parameter must be the first command, if given, after the Update Processor is called. It defines the input file as either an already assigned filecode or as a file in a user DAD.

```
type 1)  If a filecode was assigned to the file, but not /D4.
type 2)  If no filecode was assigned yet to the input file and the input
         file is a catalogued file.
```

If the file was already assigned to /D4, it is not necessary to give an !!IN command.

A non-disc file is assumed to be positioned at the first line. It is not rewound.

```
fc  A filecode already assigned to the input file.
fn  Name of the catalogued input file to be updated. The Processor assigns
    filecode /D0 to this file and reads records via DFM.
SC  The file to be updated is of type SC.
UF  The file to be updated is of type UF.
For non-disc files, the type is SC. Default = type SC.
u   Userid under which the file is catalogued. Default = :JOB USID.
d   DAD code of the userid. Default = :JOB DAD.
v   Version number of the catalogued file (from 0 to 7). Default = 0.
```

Error Messages:

```
IN NOT 1ST. COM
ILL. INP. FC.
ILL. FILE TYPE
REWIND ERR
ASG. ERR
<status> ASS. ERR
<fn> <u> <d> <type> : UNKNOWN FILENAME
"      "           : UNKNOWN USERID
"      "           : UNKNOWN DAD
      where <type> is the type of the file.
```

See also Appendix A.

Syntax with keyword parameters:

```
type 1)  !!OU FNAM=fc
type 2)  !!OU FNAM=fc,DAD=d[,TYPE={SC | UF}]
type 3)  !!OU FNAM=fn[,DAD=d][,USID=u][,TYPE={SC | UF}][,VERS=v][,RSIZ=r]
```

Syntax with positional parameters:

```
type 1)  !!OU fc
type 2)  !!OU fc,d[, {SC | UF}]
type 3)  !!OU fn[,d][,u][, {SC | UF}][,v][,r]
```

Use:

This optional command may be given anywhere in the definition phase. It defines the output file as either an already assigned filecode, or as a temporary or catalogued disc file in a user DAD which is to be assigned by UPD to a specified filecode. A disc file is rewound before the output starts.

- type 1) A filecode, but not /D1, was assigned to the output file. /D1 is scratched after UPD exits.
- type 2) No filecode has been assigned yet to the output file. In this case, fc is a filecode for a temporary file.
- type 3) The output file is written to a catalogued file. UPD assigns /D1 to this file. If updating ends successfully, the file is defined by /D4 (if type SC), or /D3 (if type UF).

No !!OU command need be given if the output file was already assigned to /D1.

- fc Already assigned filecode which will receive the updated file. If fc is accompanied by other parameters, the output is sent to a temporary disc file whose filecode is assigned by UPD in the DAD specified by d. A !!KF command must be given to keep the updated file, if necessary.
- fn Name of the catalogued disc file to which the updated file is sent. If an SMV command is in force, existing versions of the file with version numbers equal to or greater than that specified by v have their version numbers incremented by 1. If the maximum version number is 0, an existing file with the same name (and type) is deleted. An !!EN command is required to keep the updated file.
- u Userid of the catalogued file into which the updated file must be written. Default = :JOB USID.
- d DAD code of the userid containing the catalogued file into which the updated file must be written. Default = :JOB DAD.
- SC Type of the updated file.
- UF Default = the type as specified or assumed in the !!IN command.
- v Version number to be given to the updated file (0 to 7). Default = 0.
- r The output record size in characters. If the output record size is defined smaller than the input size, the records are truncated. If the output record size is defined larger than the input size, the records are right-padded with blanks.
Default: type SC: 80 characters;
 type UF: 132 characters.

Error Messages:

OU ALREADY GIVEN
ILL. FC.
NOT UF/SC TYPE
ASG. ERR (TEMPORARY FILE)
OUTPUT FILE ASG. ERR
OUT. FILE REW. ERR.
RSIZ > 132 FORBIDDEN

See also Appendix A.

Syntax with keyword parameters:

```
!!RS STNG='SA',BY='SB'[,FROM=n][,TO=m]
```

Syntax with positional parameters:

```
!!RS 'SA','SB'[,n][,m]
```

Use:

This command replaces the character string 'SA' by the character string 'SB' in the entire file, or only in that part indicated by the parameters FROM and TO. If a string contains an inverted comma, this must be enclosed as in the example on page 4-7.

- 'SA' The character string to be searched for.
- 'SB' The string which is to replace 'SA' in the line. The lengths of 'SA' and 'SB' may be different. If 'SA' occurs more than once in the same line, each occurrence will be replaced by 'SB'.
- n The first input line number at which the search for 'SA' is to start.
Default = line number 0.
- m The last input line number after which the search for 'SA' is to stop.
Default = line number containing :EOF.

Error Messages:

```
OVF ON RS ALLOC  
FROM VAL. NOT < TO VAL.
```

See also Appendix A.

Syntax with keyword parameters:

```
!!DS STNG='SA'[,FROM=n][,TO=m]
```

Syntax with positional parameters:

```
!!DS 'SA'[,n][,m]
```

Use:

Delete the specified character string 'SA' in all lines encountered, or only in that part of the file defined by the parameters FROM and TO. Lines becoming empty are not written to the output file.

'SA' The character string to be deleted.

n The first input line number at which the search for 'SA' is to start.
Default = line number 0.

m The last input line number after which the search for 'SA' is to stop.
Default = line number containing :EOF.

Error Messages:

OVF ON DS ALLOC
FROM VAL. NOT < TO VAL.

See also Appendix A.

Syntax with keyword parameters:

```
!!DE STNG='SA'[,FROM=n][,TO=m]
```

Syntax with positional parameters:

```
!!DE 'SA'[,n][,m]
```

Use:

This command deletes the line(s) containing the specified character string.

'SA' The character string to be searched for.

n The first input line number at which the search for 'SA' is to start.
Default = line number 0.

m The last input line number after which the search for 'SA' is to stop.
Default = line number containing :EOF.

Error Messages:

OVF ON DE ALLOC

FROM VAL. NOT < TO VAL.

See also Appendix A.

Syntax with keyword parameters:

```
!!IS STNG='SA' [,FROM=n][,TO=m]
```

Syntax with positional parameters:

```
!!IS 'SA' [,n][,m]
```

Use:

This command causes Update to search each line to be output for a certain character string. If the string is found, it will include in the output file, after the line containing the character string, the line(s) which is (are) given immediately after this command. The lines to be included may not start with !! or by the characters replacing !!. The lines to be included are written in the definition phase table. Their number, therefore, is limited by the memory space available for the table.

- 'SA' The character string to be searched for.
- n The first input line number at which the search for 'SA' must start.
 Default = line number 0.
- m The last input line number after which the search for 'SA' must stop.
 Default = line number containing :EOF.

Error Messages:

- OVF ON IS ALLOC
- FROM VAL. NOT < TO VAL.

See also Appendix A.

EXECUTION COMMANDS

Execution commands instruct UPD to perform a particular operation on a specified line, identified by its line number in the file, before that line is output to the output file.

Execution commands must follow definition commands, if any, and must be in ascending line number sequence.

All lines not affected by execution commands are copied to the output file after being checked against the contents of the definition phase table.

Syntax with keyword parameters:

```
!!RE LINE=n,STNG='SA',BY='SB'
```

Syntax with positional parameters:

```
!!RE n,'SA','SB'
```

Use:

This command replaces in the specified line the character string 'SA' by the character string 'SB'. A new line may be written after this command without giving an !!IL command.

n Line number of the line in which the specified character string must be replaced. The line number may be a hexadecimal number ranging from 0 through /7FFF, or a decimal number ranging from 0 through 32767.
'SA' The character string to be searched for in the specified line.
'SB' The character string to replace each occurrence of 'SA' in the specified line.

Error Messages:

```
SEQ. ERROR  
STRING NOT FOUND - CUR. LINE OUTPUT WITHOUT MODIF.  
I/O ERROR  
REPL NOT ALLOWED
```

See also Appendix A.

Syntax with keyword parameters:

```
!!DL FROM=n[,TO={m | EOF}]
```

Syntax with positional parameters:

```
!!DL n[,{m | EOF}]
```

Use:

This command deletes the line with line number n, or the lines from n to m, both included, or all lines up to the end of file.

New lines may be input immediately after this command is executed, without using an !!IL command.

- n The line number of the line to be deleted, or the line number of the first line to be deleted when the parameter TO is also specified. The line number is a hexadecimal number ranging from 0 through /7FFF or a decimal number ranging from 0 through 32767.
- m All lines, from line number n to m, both included, are deleted.
- EOF All lines from n (included) to :EOF are deleted.
Default: Only the line specified in n is deleted.

Error Messages:

```
SEQ. ERROR  
INP. FILE OVF  
OUT. FILE OVF  
I/O ERROR FC=/<fc>, ST=<yyyy>
```

See also Appendix A.

!!IL

Insert Line(s)

!!IL

Syntax with keyword parameters:

```
!!IL [AFTR=x]
```

Syntax with positional parameters:

```
!!IL [x]
```

Use:

This command inserts, after the line number specified, all the lines which may now be input and which do not start with !! or its replacements. Lines are inserted until a valid UPD command is given. The command without parameter can be used to include lines before line number 0 of the file, if it is the first executable command.

The line number may be a hexadecimal number ranging from 0 through /7FFF, or a decimal number ranging from 0 through 32767.

x The line number after which new lines are to be added. If omitted, lines are inserted at the current point in the file.

Error Messages:

```
SEQ. ERROR  
I/O ERROR FC=/fcILL. AFTR VALUE  
INP. FILE OVF  
OUT. FILE OVF
```

See also Appendix A.

Syntax with keyword parameters:

```
!!JN [AFTR=x],AUXI=fc[,FROM=n][,TO=m]
!!JN [AFTR=x],AUXI=fn[,TYPE={SC | UF}][,USID=u][,VERS=v][,DAD=d]
      [,FROM=n][,TO=m]
```

Syntax with positional parameters:

```
!!JN [x],fc[,n][,m]
!!JN [x],fn[,{SC | UF}][,u][,v][,d][,n][,m]
```

Use:

This command permits the user to include, after the specified line number:

- a part of another file;
- a part of the file being updated (disc file only).

The auxiliary file is defined by either an already assigned filecode or by a file in a user DAD.

More than one !!JN command may be given if the auxiliary file is a disc file, as the auxiliary file is rewound when a !!JN command is given. If the auxiliary file is a non-disc file, only one !!JN command may be given, as these files are not rewound by the system.

When this command has been given, new lines may be included without giving an !!IL command, as the processor is still pointing to the line specified by x.

- x A hexadecimal line number ranging from 0 through /7FFF, or a decimal number ranging from 0 through 32767, after which a part of or the entire auxiliary file may be written.
Default = the lines of the auxiliary file are included at the current position in the file, or before line number 0 if !!JN is the first execution command given.
- fc Filecode already assigned to the auxiliary file.
- fn Name of the catalogued auxiliary file.
- d DAD code of the userid containing the catalogued file.
Default = :JOB DAD.
- u Userid containing the catalogued file. Default = :JOB USID.
- v Version number of the catalogued file. Default = 0.
- TYPE File type SC or UF of the auxiliary file. The auxiliary file need not be of the same type as the file to be updated, nor the output file.
Default = input file type.
- n Line number of the first line to be included in the file to be updated. Default = line number 0 of the auxiliary file.
- m Line number of the last line to be included in the file to be updated.
Default: the last line before :EOF.

Error Messages:

ILL. AUXI TYPE
ASG. ERR
AUXI. REW. ERR
AUXI IO ERROR
IO ERROR FC=/**<fc>**,ST=**<yyyy>**
SEQ. ERROR
INP. FILE OVF
AUXI. FILE OVF
LINE NO. > /7FFF
TO < FROM
ILL. AFTR/FROM/TO VAL.

See also Appendix A.

Note: Lines copied from the auxiliary file will not be checked against the information in the definition phase table.

Syntax !!EX

Use:

This command causes the immediate exit from the Update Processor to BCP, with severity code /60.

The temporary output file /D3 or /D4 is deleted. If the output file is written to a catalogued file, this file is closed and is not changed.

Filecodes remain assigned as they were before Update started. This command is to be used when updating cannot be terminated normally.

Error Messages:

OUT. FILE CLOSE ERR.

See also Appendix A.

Syntax !!EN

Use:

This command, or the command !!KF, must be the last command given in the UPD command stream, except when !!EX was given previously.

All remaining lines up to :EOF on the input file are copied onto the output file, after having been submitted to the information in the definition phase table.

The command !!EN must be used when:

- A type 1 !!OU command, or no !!OU command, was specified (i.e. /D1 is assigned to the output file when Update starts and /D3 and /D4 are the final Update filecodes).
- The updated file need not be catalogued.
- The output file was assigned to a non-disc device.
- The output file is written into a catalogued file.

After execution of the !!EN, control is returned to BCP. If the output is written to a temporary file, the file is not destroyed when UPD exits to BCP.

Error Messages:

OUT. FILE ASG. ERR.
OUT. FILE CLOS. ERR.
IO ERROR FC=<fc>,ST=<yyyy>
SEQ. ERROR
OUT. FILE OVF

See also Appendix A.

Syntax with keyword parameters:

```
!!KF [FNAM=fn][,TYPE={SC | UF}][,USID=u]
```

Syntax with positional parameters:

```
!!KF [fn][,{SC | UF}][,u]
```

Use:

This command terminates successful updating and catalogues the updated file in the relevant DAD.

All remaining lines on the input file are copied onto the output file after having been checked against the information in the definition phase table.

The DAD in which the output file is written is specified by the DAD parameter of type 2 and type 3 !!OU commands. For a type 1 !!OU command, the DAD was determined when the filecode fc was assigned.

If no !!OU command was given for a non-disc device or a catalogued file, an error message is output. Use in those cases the !!EN command.

When the !!KF command has been executed, control is returned to BCP.

- fn The name to be given to the updated file. This name is placed in the directory for userid u of the :JOB USID.
Default: The name of the input file. In this case the first record of the input file must be an IDENT record.
- u Userid of the directory in which the updated file must be entered.
Default = :JOB USID.
- SC The type of the file to be catalogued.
- UF
Default = type SC, except when:
 - type UF was specified in !!OU
 - type UF was specified in !!IN
 - no !!OU given
 - type UF was specified in !!IN and no TYPE parameter in !!OU command.

Error Messages:

```
IO. ERR. FC= /<fc>, ST= <yyyy>
ILL. OUT. IDENT
FILE TYPE ER.
FILE NOT SC
OUT. FILE ALR. CAT.
OUT. FILE CLOS ERR.
REW. ERR. OVF
KEEP FILE ERR.
```

See also Appendix A.

GENERAL COMMANDS

General commands may be given at any moment in updating.

!!CC

Change Special Characters

!!CC

Syntax with keyword parameters:

!!CC SPEC=cc

Syntax with positional parameters:

!!CC cc

Use:

This command allows to replace the standard two exclamation marks !! in front of the UPD commands by two other convenient equal characters.

All following UPD commands must commence with the newly defined characters until a new !!CC command is given or until the end of updating.

A new CC command must be preceded by the last two special characters defined.

Error Messages:

SPEC. CHAR. DIFFERENT.

See also Appendix A.

Syntax with keyword parameters:

```
!!CI [FCOD=fc]
```

Syntax with positional parameters:

```
!!CI [fc]
```

Use:

This command changes the input stream of commands to fc. If no parameter is supplied, the input resumes from its original source - either filecode /EO or from a catalogued procedure.

Error Messages:

```
INVALID FILE CODE  
REWIND ERROR ON F. CODE GIVEN BY CI
```

Syntax with keyword parameters:

!!HL [CMND=<cmd>]

Syntax with positional parameters:

!!HL [<cmd>]

Use:

This command lists the syntax of one or all Update commands on filecode /01.

<cmd>: one of the Update commands, without special characters. If omitted, the syntax of all commands is listed.

Error Messages:

INVALID COMMAND - <cmd> is not a valid Update command.

See also Appendix A.

Syntax with keyword parameters:

```
type 1) !!LF FNAM=fc[,STNG='SA'][,FROM=n][,TO=m]
type 2) !!LF FNAM=fn[,TYPE={SC | UF}],USID=u[,DAD=d][,VERS=v][,STNG='SA']
```

Syntax with positional parameters:

```
type 1) !!LF fc[, 'SA'][,n][,m]
type 2) !!LF fn[, {SC | UF}][,u][,d][,v][, 'SA'][,n][,m]
```

Use:

This command permits the user to list, at any moment in updating, part or all of the file to be updated.

```
type 1) A filecode was assigned to the file to be listed, but not /D4.
type 2) No filecode is yet assigned to the input file, and the input file
is a catalogued file.
```

```
fc      A filecode already assigned to the file.
fn      Name of the catalogued file to be listed.
SC      Type of the file to be listed. Default = SC.
UF
u       Userid of the catalogued file. Default = :JOB USID.
d       DAD code of the userid u. Default = :JOB DAD.
v       Version number of the catalogued file. Default = 0.
'SA'   Only lines containing this string will be listed.
       Default = All lines are listed between FROM and TO or from line 0
       to :EOF.
n       The first line which must be listed. Default = line number 0.
m       The last line to be listed. Default = :EOF.
```

Error Messages:

```
FROM VAL. NOT < TO VAL.
ILL LIST FILE FC
ILL. FILE PARAM.
IO ERROR
```

See also Appendix A.

EXAMPLESExample 1

Read a file from the punched tape unit (FCD /E1,PR20). Update the file and write the output to the cassette unit (FCD /3,TK05).

```
:JOB USID=SYSTEM

DATE: FEB 23 78 // TIME : 00H.00M.00S.

** BCP RELEASE 18.01.78
UPD
  UPDATE REL 06 (01 06 76)
!!IN FNAM=/E1
!!OU FNAM=/3
!!RS STNG='A1',BY='A2'
!!EN
```

Example 2

The file to be updated is a catalogued file which is defined in the !!IN command. The updated file is written to the cassette unit (FCD /3,TK05).

```
:JOB USID=SYSTEM

DATE: FEB 23 78 // TIME : 00H.00M.00S.

** BCP RELEASE 18.01.78
UPD
  UPDATE REL 06 (01 06 76)
!!IN FNAM=M:CC1,TYPE=SC,USID=MWSC2,DAD=/F2
!!OU FNAM=/3
!!IL 35
I *THIS MODULE UPDATES THE SYNTAX OF THE LINE
!!EN
```

Example 3

Read the file to be updated from the punched tape unit (FCD /E1,PR20). Write the updated file to a temporary file (/20) and then catalogue the file in DAD /F2 and USID MWSC2 (:JOB USID and DAD).

```
:JOB USID=MWSC2,DAD=/F2

DATE: FEB 23 78 // TIME : 00H.00M.00S.

** BCP RELEASE 18.01.78
UPD
  UPDATE REL 06 (01 06 76)
!!IN FNAM=/E1
!!OU FNAM=/20,DAD=/F2
!!DL FROM=17
D *THIS MODULE IS ACTIVATED WHEN TYPING IN THE COMMAND 'CC'.
!!KF FNAM=M:CC1,TYPE=SC
```

Example 4

A file, catalogued in DAD /F2 and USID MWSC2, is assigned to /E1. The updated file is written to temporary file /20, and then catalogued with an !!KF command in DAD /F3.

Since no FNAM parameter is specified in the !!KF command, the name in the IDENT statement of the updated file is taken as the name for the catalogued file.

```
:JOB USID=MWSC2,DAD=/F3
```

```
DATE: FEB 23 78 // TIME : 00H.00M.00S.
```

```
** BCP RELEASE 18.01.78
ASG FCOD=/E1,FNAM=M:CC1,DAD=/F2,USID=MWSC2,TYPE=SC
UPD
  UPDATE REL 06 (01 06 76)
!!IN FNAM=/E1
!!OU FNAM=/20,DAD=/F3
!!IL AFTR=26
I \LDKL\A13,/3737\SET CHARACTER COUNT
!!KF
LIB
```

```
DATE: FEB 23 78 // TIME : 00H.00M.00S.
LIBRARIAN REL 06 (01 06 76)
PRD
```

```
      LABEL = MWSC          DATE = 02 01 74          1215PACK NBR = 0102
```

```
DAD  MASDC2          LIBRARY DIRECTORY OF USER  MWSC3
```

FILE NAME	TYPE	VERS	DATE	ADDR	ORG	ATTRIBUTES
SORT	SC	1	23 02 78	0020	NC	SH
SORT	SC	0	23 02 78	0030	NC	SH
M:CC	SC	0	23 02 78	0040	NC	SH

Example 5

A catalogued file M:CC1 in DAD /F2 must be updated. Lines 35 through 71 have to be inserted after line 12 of the same file. The updated file is written to the catalogued file M:CC in DAD /F3. A !!JN command can be used.

```
:JOB USID=MWSC3,DAD=/F3
```

```
UPD
  UPDATE REL 06 (01 06 76)
!!IN FNAM=M:CC1,TYPE=SC,DAD=/F2,USID=MWSC2
!!OU FNAM=M:CC,TYPE=SC
!!JN AFTR=12,AUXI=M:CC1,TYPE=SC,USID=MWSC2,DAD=/F2,FROM=35,TO=71
!!EN
```

Example 6

The input file is the same as the output file. The Processor outputs an error message.

```
UPD
  UPDATE REL 06 (01 06 76)
  !!IN FNAM=M:CC,TYPE=SC
  !!OU FNAM=M:CC,TYPE=SC
  !!EN

UPD=**** IN FILE = OUT FILE
EXIT CODE = 00E0
  PSW=/FCC5,A1 =/0000,A2 =/0008,A3 =/0000,A4 =/00FF,A5 =/12AA
  A6 =/0A98,A7 =/00E0,A8 =/0424,A9 =/1C84,A10=/0000,A11=/0000
  A12=/0000,A13=/0000,A14=/0688,FR1=/0000,FR2=/0000,FR3=/0000
```

Example 7

In this example several methods of using the !!IN and !!OU commands are given.

In the first example !!IN is a catalogued file which is output to the defaulted !!OU file (/D4). /D4 is then used as the input for the Assembler, where the PROG parameter in the OPT statement must be /D4.

```
UPD
  UPDATE REL 06 (01 06 76)
  !!IN FNAM=M:CC,TYPE=SC
  !!IL AFTR=26
  I \LDKL\A12,89
  !!EN
ASM

DATE: FEB 23 78 // TIME : 00H.00M.00S.

  ASM VERSION NUMBER 771101
  OPR PROG=/D4,LIST=YES

ASM: OPTION STATEMENT MISSING
  OPT PROG=/D4,LIST=YES
```

The assembled file was not yet correct. The file is still on /D4 and UPD is called again. No !!OU command is given, as the output must be written again to /D4.

```
UPD
  UPDATE REL 06 (01 06 76)
  !!IL AFTR=27
  I \ABL\SORT
  !!EN
ASM

DATE: FEB 23 78 // TIME : 00H.00M.00S.

  ASM VERSION NUMBER 771101
  OPT PROG=/D4,LIST=YES
```

The final updated output is:

00025			M:CC	EQU	*
00026	0000	85A0		LDKL	A13,0
	0002	0000			
00027	0004	84A0		LDKL	A12,89
	0006	0059			
00028	0008	8F20		ABL	SORT
	000A	0000	F		
00029	000C	F6A1		CF	A14,M:MESS
	000E	0000	X		

Since the file is on /D4, the Assembler may now be called with PROG=/D4.