

Vol. 2: PROGRAMMING INFORMATION
Part 1: PROGRAMMING LANGUAGES
Section 5: MTFORTRAN Subroutines

Contents

	Page
Chapter 1: INTRODUCTION	
1.1 Purpose	1
1.2 Form of Distribution	1
1.3 Method of Use	1
1.4 Configuration	1
Chapter 2: FUNCTIONS	
2.1 Principles of Use	3
2.1.1 Formats on Tape	3
2.1.2 File Protection and Initialisation	3
2.1.3 Closing a file	3
2.1.4 Transfer of Data	3
2.1.5 Declaration of GLOBAL statement	3
2.2 MTCHECK	3
2.3 MTOPEN	5
2.4 MTWRITE	7
2.5 MTREAD	8
2.6 MTCLOSE	8
Chapter 3: STORE USED	11
Chapter 4: TIME TAKEN	13

and I am still learning how to do
it, but I am getting better at it.
I am learning that it takes time.

Conclusion

It is clear that the model
is not yet complete, but
it has been developed
to the point where it can
be used effectively.

The model is currently being
used to predict the behavior of
the system under different
conditions, and the results
are promising.

The model is currently being
used to predict the behavior of
the system under different
conditions, and the results
are promising.

The model is currently being
used to predict the behavior of
the system under different
conditions, and the results
are promising.

Chapter 1: INTRODUCTION

1.1. Purpose

The MTFORTRAN subroutines enable a programmer using 903 Basic FORTRAN to read and write files and blocks of data to and from magnetic tape (The 905 FORTRAN user should use the 905 MTFILE routines).

1.2 Form of Distribution

The subroutines are distributed on a single paper tape; 903 MTFORTRAN. This tape being in 900 SIR assembly code form. They are also distributed on the FAS systems magnetic tape.

1.3 Method of Use

The procedures are added to the users FORTRAN program as pretranslated library subroutines, using the appropriate method of loading and/or conversion to relocatable binary.

The names of any of the routines called within a program unit must be included in the GLOBAL list at the head of that unit.

1.4 Configuration

A basic 903 computer with at least one 9KHz magnetic tape handler, and 8K or 16K store.

REPORTS AND

ANNUAL REPORTS AND ACCOUNTS

FOR THE YEAR ENDED 31ST MARCH, 1903.

THE ANNUAL REPORTS AND ACCOUNTS FOR THE YEAR ENDED 31ST MARCH, 1903,

ARE HEREBY PUBLISHED IN ACCORDANCE WITH THE REQUIREMENTS OF THE COMPANIES ACT, 1903.

THE ANNUAL REPORTS AND ACCOUNTS FOR THE YEAR ENDED 31ST MARCH, 1903,

ARE HEREBY PUBLISHED IN ACCORDANCE WITH THE REQUIREMENTS OF THE COMPANIES ACT, 1903.

THE ANNUAL REPORTS AND ACCOUNTS FOR THE YEAR ENDED 31ST MARCH, 1903,

ARE HEREBY PUBLISHED IN ACCORDANCE WITH THE REQUIREMENTS OF THE COMPANIES ACT, 1903.

THE ANNUAL REPORTS AND ACCOUNTS FOR THE YEAR ENDED 31ST MARCH, 1903,

ARE HEREBY PUBLISHED IN ACCORDANCE WITH THE REQUIREMENTS OF THE COMPANIES ACT, 1903.

Chapter 2: FUNCTIONS

2.1 Principles of Use

The MTFORTRAN procedures assume that files are read and written according to the formats described in 900 Manual, Section 2.7.1. Thus it is possible to interchange files written by SIR, ALGOL and FORTRAN programs.

2.1.1. Formats on Tape

All magnetic tape reels to be processed by standard software must have a header block and the first two words of every data block; written in standard format (see Volume 2.7.1).

2.1.2 File Protection and Initialisation

Before data can be written onto or read from a tape the file specified on the tape must be 'opened' by the procedure MTOPEN. Unless it is a SCRATCH tape (i.e. the header name is SCRATCH) the header block name must correspond exactly to the name used in MTOPEN. This procedure guards against unauthorised overwriting or reading of files. The reel must be prepared initially by using the utility program MTINIT, before any data can be written onto it.

2.1.3 Closing Files

Before finishing a program, particularly when the tape reels are to be removed after the run, all files used by the program must be closed by the procedure MTCLOSE. If a file has been written or updated, it must be closed for writing before the end of the program.

2.1.4 Transfer of Data

Data is read and written in blocks, corresponding to the physical blocks recorded by the hardware. These blocks are written from or read into buffers in core store, which are specified by the user as integer or real arrays within his FORTRAN program. Data is written and read by means of procedures MTWRITE and MTREAD respectively.

Character strings may be transferred between the buffer and another medium by the subroutines INSTRING and OUTSTRING (see 2.1.7).

2.1.5 Declaration by GLOBAL statement

Users of Basic 903 FORTRAN must declare any of the MTFORTRAN subroutines actually used within a program unit, by including their names in the GLOBAL statement for that unit.

2.2. MTCHECK

MTCHECK (NH, NB, ISTAT)

MTCHECK is an integer function subprogram used to check the operation of the other magnetic tape FORTRAN procedures. It must be used after each of the other procedures, before any other reference to any of the tapes, and

its exact effect is defined in the individual procedure descriptions. The general effect is as follows:

NH is the handler number required in the range 0 to 3. The value of NH must correspond to that used in the previous procedure, except when checking the completion of a rewind after MTCLOSE.

If the previous procedure was successful and normal MTCHECK returns the value +1. In this case parameter NB will hold the block number of the last block read or written. Parameter NSTAT will hold the status word read from the magnetic tape controller on completion of the operation.

If the previous procedure was not successfully completed, MTCHECK returns the value -1. In this case parameter B holds an error number. Parameters NSTAT again holds the status word, which will indicate the most recent hardware error state.

Values of NB under error conditions are as follows:

NB = 1	Repeated hardware error, failure after 20 attempts to read or write a block.
NB = 2	File not opened by MTOPEN as names not compatible.
NB = 3	Specified block for reading cannot be found.
NB = 4	Handler in Manual or not available.
NB = 5	Instruction rejected as 'do nothing', due to hardware error or manual interference.
NB = 6	Attempt to write or read more than one block after the end of tape warning marker has been detected.
NB = 7	File not opened for reading or writing before an attempt to read or write respectively.
NB = 8	Write permit ring not in when attempting to write.
NB = 9	Long block. The block read from magnetic tape is longer than the buffer allocated.

The following cases are not errors, but MTCHECK returns the value +0 as special action may be required:

NB = 0 The end of Tape warning marker has been detected while reading or writing the last specified block. If writing a file, the file should be closed for writing, as no further data blocks are allowed on that reel.

- NB = -1 The block specified to be read is label block (other than EOF (end of file) block).
- NB = -2 The block specified to be read is an EOF (End of file) block.
- NB = -3 The handler is still busy rewinding after a call of MTCLOSE.

2.3 MTOPEN

MTOPEN (NH, IA, NT)

This procedure must be used to 'open' a file on a magnetic tape reel, before reading or writing data. When used it must be followed by a call of MTCHECK.

NH is the handler number required. The actual parameter must be an integer in the range 0 to 3. Note that a program will be easier to adapt if all references to a handler are by an integer variable, which may be set at the beginning of the program.

The name of the file to be opened must be specified. This is normally specified by writing a numbered FORMAT statement immediately following the call of MTOPEN. The first item in this format statement must be an H description, and when the program calls MTOPEN this field must contain the name of the file. See examples below.

IA is an integer array used as a buffer. It must be at least 56 words long. The array elements are referred to as IA(1), IA(2), . . . to IA(56), assuming that the array is declared as IA(n). Once MTOPEN and MTCHECK have been used, this array may be used for other purposes.

NT is a parameter indicating the type of operation required. The types of operation are:

- NT = -1 Open file for reading only. The name of the file must correspond exactly to the name given. The tape is positioned ready to read the first data block.
- NT = 0 Open file for updating. The name of the file must correspond exactly to the name supplied. The tape is positioned ready to read or write the first data block.
- NT = 1 Open file for writing. The original name on the file must correspond to that given, or the reel must hold a scratch file, with name SCRATCH. IA(29) to IA(56) must be set to appropriate values

If the file is opened for updating the existing header block is checked but not overwritten. If information is to be added to the end of the file

the program should read forwards until the EOF block is detected, go back one block, and start writing.

If the file is opened for writing, a new header block is written, and a complete new file must be written.

In each case, after the call of MTCHECK following MTOPEN the original header block read from the tape will be left in the buffer array A, in the first 28 elements; IA(1) to IA(28). When opening for writing the header block is written from locations IA(29) to IA(56). These locations should all be set to zero (or to appropriate values required for the header block locations 0; to 27,) before MTOPEN is used.

As described above, the file name is normally specified by a FORMAT statement immediately following the call of MTOPEN. This statement must start with an H descriptor, and there must not be any separator (i.e. ,/z) before the H descriptor. The H descriptor may specify up to 12 characters, further characters will be ignored. The H descriptor may be followed by any separators and fields. The FORMAT statement must be numbered.

If MTOPEN is not followed by such a statement, the file name is taken from the array IA; elements IA(31) to IA(34) of an integer array should hold 12 characters, packed 3 to a word in 6 bit internal code. Any non-significant characters should be zero. The user's program must set up these elements before calling MTOPEN.

The following examples show the use of the FORMAT statement to specify a file name.

```
C EXAMPLE 1
GLOBAL, EX1, MTOPEN, MTCHECK]
DIMENSION IA(56)
CALL MTOPEN (2, IA, 1)
999 FORMAT (12HPAYROLLFILE2)
C THIS FORMAT STATEMENT SPECIFIES FILE NAME FOR
C READ OR WRITE STATEMENT TO BE USED WITH MTOPEN.
C IT IS NOT USED BY ANY READ OR WRITE STATEMENT,
C BUT IT MUST HAVE A STATEMENT NUMBER.
IF (MTCHECK(2,NB,STAT))1,2,3
C EXAMPLE 2
GLOBAL , EX2, MTOPEN, MTCHECK]
DIMENSION IA(56)
7 WRITE (3,10)
10 FORMAT (/33H TYPE FILE NAME AND HANDLER NUMBER)
READ (3,11) NH
CALL MTOPEN (NH, IA, 1)
11 FORMAT (12H123456789012,I2)
IF (MTCHECK (NH,NB,ISTAT)) 4,5,6
4 WRITE (3,12) NB,ISTAT
12 FORMAT (5HERROR,I3,6HSTATUS,I6)
GOTO7
```

2. 4 MTWRITE

MTWRITE (NH, IA, NP)

This procedure is used to write a block of data (or a label block) onto magnetic tape. When used it must be followed by a call of MTCHECK.

NH is the handler number required (in the range 0 to 3).

IA is an integer array which must contain the data block to be written. The first 2 words of the array must not contain significant data, as they will be overwritten when the block is written onto tape.

NP must be set to the number of words to be written in the block. This number includes the 2 locations at the lower end of the array, e.g. if JAA(1) to JAA(200) are to be output on handler 2, the call might be:

```
CALL MTWRITE (2,JAA,200);
```

JAA(1) and JAA(2) will be overwritten. This example assumes that JAA is declared as DIMENSION JAA (300) say.

After the call of MTWRITE array IA must not be referred to or altered until MTCHECK has been called.

The block will be written onto the tape, starting at the current position of the tape under the Read/Write heads. Writing should normally be in sequence along the tape from the header block written by MTOPEN. Each call of MTWRITE leaves the tape positioned ready for the next call. When all blocks of a file have been written MTCLOSE should be used to close for writing. The call of MTCHECK following each call of MTWRITE will give the block number of the block just written, and if necessary the program may keep a record of certain block numbers, so that they can be conveniently found by MTREAD.

If blocks are to be written from a different position, MTREAD should be used to position the tape. As soon as a block is written all other blocks further along the tape should be considered unreadable, (though in exceptional circumstances they may be found using MTREAD to read a specific block number). It is not possible to overwrite odd blocks in the middle of an existing file and preserve the rest of the file. If random blocks are to be updated, the whole file must be copied onto a new tape reel, adding or deleting the necessary blocks as they appear in the copying sequence.

Up to 20 attempts are made to write the block specified, and small patches of bad tape are erased over before the block is written.

If a label block is to be written, parameter NP should be given value (number of words - 131071 - 1).

2.5 MTREAD.

MTREAD (NH, IA, N)

This procedure is used to find and read a specified block on a magnetic tape reel. Since all blocks are numbered, a block with a given number may be found, searching forwards or backwards along the tape.

NH is the handler number required (in the range 0 to 3).

IA is an integer array which is used as a buffer for input of blocks. The whole array is used. After reading and successfully checking with MTCHECK the block specified by N will be held in IA, with the first two words holding special information inserted by MTWRITE when the block was written.

N parameter specifies the block to be found and read. If N = 0 the next block in sequence is read. If N is positive, non-zero then the block with number N is found.

The array IA should not be referred to until after the call of MTCHECK following MTREAD. If the actual block specified to be read is a label block then MTCHECK returns value +0. However, the block will still be read correctly into array IA, with the block number in the first location of IA.

The search for a numbered block will start with a rewind or a number of backspace movements if necessary, then will continue forward till the block is found. The search forward is not fundamentally quicker than reading repetitively with MTREAD (NH, IA,), unless there are a number of bad patches on the tape.

Up to 20 attempts will be made to read the actual block specified, other blocks are ignored. The actual block length (number of words) will be found in the second location of IA, unless a label is read, when this location holds the length - 131072.

2.6 MTCLOSE

MTCLOSE (NH, NT)

This procedure is used to 'close' a magnetic tape file after reading or writing is complete. Files that have been written or updated must be closed for writing once the last block has been added. In general, at the end of a program run, and always before a reel is removed, any reels used by the program should be closed.

MTCLOSE may also be used to rewind a tape without closing the file.

NH is the handler number required, in the range 0 to 3.

NT is a parameter giving the type of operation required as follows:

- NT = 1 Close file for writing. Write an End of File label and rewind tape.
- NT = 2 Close file for writing and reading. Write an End of File label and rewind.
- NT = 3 Close file for reading only. Rewind tape (Only use if file not opened for writing).
- NT = 0 Rewind tape without closing file.

After closing for writing ($T = 1$ or $T = 2$), MTCHECK must be used before any other reference to any of the tapes. After this (and after any other call of MTCLOSE) MTCHECK may be used at any time to check whether the rewind operation is complete.

Chapter 3: STORE USED

Approximate core store used:

MTCHECK, MTOPEN, MTREAD, together = 600 words

MTCLOSE 65 words

MTWRITE 110 words

Total: 775 words

Chapter 4: TIME TAKEN

The time taken for these procedures is governed almost entirely by the tape movement times, and it is virtually independent of the processor speed.

Approximate times for 9kHz tape, assuming no re-reads or re-writes due to tape errors, may be calculated as follows, if W is the number of words in a block:

MTWRITE	$18 + 0.33W$ millisec.
MTOPEN	28 millisec. plus rewind time, to open for reading or updating
	80 millisec. plus rewind time, to open for writing.
MTCLOSE	20 millisec. to close for writing. 1 millisec. to close for writing.
MTCHECK	0.8 millisec (100 microsec. on 905, 1μS store)
MTREAD	$18 + 0.33 W$ millisec. to read next block.

If MTREAD is used to search for a given block number N and L is the number of the last block read or written, then the time taken in seconds is very approximately given by:

```
if N > L then (N-L) * (18 + 0.33 W)/1000  
else if N ≤ L/2 then (L/150) * (W/66 + 0.75) + (N/1000) * (18 + 0.33 W)  
else (L - N + 2) * (18 + 0.33 W)/1000;
```

In this case W is the average number of words in a block.