

# ELLIOTT

# 9000

Volume 2: PROGRAMMING INFORMATION  
Part 7: MAGNETIC TAPE ROUTINES  
Section 9: MAGNETIC TAPE UTILITY ROUTINES

	Contents	Page
Chapter 1: INTRODUCTION		
1.1	General .. .. .	1
1.2	Routines Provided .. .. .	1
1.3	Program Control using teleprinter or paper tape ..	1
1.3.1	Entry Points .. .. .	1
1.3.2	Character Set .. .. .	2
1.3.3	Input of Numbers .. .. .	3
1.4	Error Procedures .. .. .	3
1.5	Configuration .. .. .	4
Chapter 2: MTINIT		
2.1	Introduction .. .. .	5
2.1.1	Purpose .. .. .	5
2.1.2	Form of Distribution .. .. .	5
2.1.3	Method of Use .. .. .	5
2.1.4	Output .. .. .	5
2.2	Functions .. .. .	5
2.2.1	Entry Points .. .. .	5
2.2.2	Operation .. .. .	5
2.3	Errors .. .. .	6
2.4	Store Used .. .. .	6
2.5	Timing .. .. .	6

Chapter 3: MTLIST

3.1	Introduction	..	..	..	..	..	..	..	..	7
3.1.1	Purpose	..	..	..	..	..	..	..	..	7
3.1.2	Form of Distribution	..	..	..	..	..	..	..	..	7
3.1.3	Method of Use	..	..	..	..	..	..	..	..	7
3.1.4	Output	..	..	..	..	..	..	..	..	7
3.2	Functions	..	..	..	..	..	..	..	..	7
3.2.1	Entry Points	..	..	..	..	..	..	..	..	7
3.2.2	Operation	..	..	..	..	..	..	..	..	7
3.2.3	Format	..	..	..	..	..	..	..	..	9
3.2.3.1	Header Block	..	..	..	..	..	..	..	..	9
3.2.3.2	Label Block	..	..	..	..	..	..	..	..	9
3.2.3.3	Data Blocks	..	..	..	..	..	..	..	..	10
3.3	Errors	..	..	..	..	..	..	..	..	10
3.4	Store Used	..	..	..	..	..	..	..	..	10
3.5	Timing	..	..	..	..	..	..	..	..	10

Chapter 4: MSDUMP

4.1	Introduction	..	..	..	..	..	..	..	..	11
4.1.1	Purpose	..	..	..	..	..	..	..	..	11
4.1.2	Form of Distribution	..	..	..	..	..	..	..	..	11
4.1.3	Method of Use	..	..	..	..	..	..	..	..	11
4.1.4	Output	..	..	..	..	..	..	..	..	11
4.2	Functions	..	..	..	..	..	..	..	..	11
4.2.1	Entry Points	..	..	..	..	..	..	..	..	11
4.2.2	Operation	..	..	..	..	..	..	..	..	11
4.3	Errors	..	..	..	..	..	..	..	..	11
4.4	Store Used	..	..	..	..	..	..	..	..	11
4.5	Timing	..	..	..	..	..	..	..	..	11

Chapter 5: MTCALL

5.1	Introduction	..	..	..	..	..	..	..	..	14
5.1.1	Purpose	..	..	..	..	..	..	..	..	14
5.1.2	Form of Distribution	..	..	..	..	..	..	..	..	14
5.1.3	Method of Use	..	..	..	..	..	..	..	..	14
5.1.4	Output	..	..	..	..	..	..	..	..	14
5.2	Functions	..	..	..	..	..	..	..	..	14
5.2.1	Entry Points	..	..	..	..	..	..	..	..	14
5.2.2	Operation	..	..	..	..	..	..	..	..	14
5.3	Errors	..	..	..	..	..	..	..	..	15
5.4	Store Used	..	..	..	..	..	..	..	..	15
5.5	Timing	..	..	..	..	..	..	..	..	15

Printed in England  
English Electric Computers Limited.

## Chapter 1 . INTRODUCTION

### 1.1 General.

These routines are designed to perform the most common and essential jobs in a magnetic tape system. The input/output routines are identical throughout and the programs incorporate all the safeguards in the standard software subroutines.

### 1.2 Routines Provided.

Initialise file, dump program, retrieve program, list file contents.

### 1.3 Program Control using teleprinter or paper tape.

This section deals with the conventions using teleprinter or paper tape to put instructions or information to these programs. The input/output procedures are identical and standard procedures, therefore apply throughout.

#### 1.3.1 Entry Points.

Every utility routine has two entry points:

0; for input via teleprinter.

1; for input via paper tape.

(0; is the first location of the program and 1; is the second location) Thus if the program is assembled at 32 onwards, the entry points will be 32 and 33 for teleprinter and paper tape control respectively. The user will normally find it convenient to assemble one or more versions of the utility routines at convenient positions in the store, and dump these onto paper tape or magnetic tape, using either T22 (Volume 2.3.1) or MSDUMP.

The input/output select switches should be set to AUTO. When paper tape is used the programs will stop at some points instead of continuing to read. This enables the operator to check any output, where this happens it is mentioned in the specification of the routine.

1: 3.2 Character Set.

Character	Action
<p>blank, erase, carriage return.</p> <p>Lower case letters</p> <p>Halt</p> <p>Seperators (S) (L) (T) (Space, newline, Tab)</p> <p>Terminators: , (Colon, comma)</p> <p>Program Terminate )</p> <p>Cancel / (slash)</p> <p>Odd Parity Characters and characters not in the internal character code set</p> <p>All other characters in the internal code set.</p>	<p>Always Ignored.</p> <p>Treated as upper case letters.</p> <p>Causes wait; enter at 9, to continue.</p> <p>Ignored up to first significant character, then treated as other characters, tab is treated as space.</p> <p>Colon and Comma are terminators. Ignored on initial run-in until first significant character, then terminate number or string input. When they have different effects this is mentioned in the routine specification.</p> <p>When the close bracket is the first significant character of an input item it is treated as either a program terminator or an error (see routine). If not first character it is treated as any other.</p> <p>Slash (/) is the cancel sign, the program ignores anything already input for the current item, prints asterisk, linefeeds; and recommences input procedure.</p> <p>Invalid; error output (see 1.4)</p> <p>Accepted for input in a name, only digits are accepted when inputting a number.</p>

### 1.3.3 Input of Numbers.

Only positive integers can be input to these routines and no sign is necessary. The number is terminated by a separator or terminator but input of characters will continue until a terminator is found. Any character other than digits, separators and terminators will give an error as will a separator placed within a number (e. g. 1 (S) 2). Numbers input greater than 131,071 will give an error.

Example: (L) (S) 1234, is an acceptable number input.

### 1.3.4 Input of alphanumeric strings.

Any characters may be input providing they have no defined effect. The input will only be terminated by either a comma or colon even when there is a defined limit to the length of a string in which case the excess will eventually be ignored.

Example: (L) (S) ABC\*1234, will be accepted as the 6 character program name ABC\*12. However the user is advised to normally restrict names to letters and figures only.

### 1.4. Error Procedures.

Error messages given are normally in standard format i. e:  
\*AAA n m.  
where AAA represents 3 characters defining the program routine in which the error occurs.

n is the status word of the magnetic tape handler in octal form (unless otherwise specified).

m is an error number unless otherwise specified, printed in 5 digit decimal form.

As far as possible these have standard value as follows:

- 1) Programme Routine  
MTI refers to MTINIT  
MTC refers to MTCALL  
MSD refers to MSDUMP  
MTL refers to MTLIST
- 2) The status bits have the following meaning.

Bit	Set by
1	Handler busy
2	Handler in manual state
3	Missed data transfer
4	Parity Error
5	Short block
6	Long block
7	Write permit ring fitted
8	Beginning of tape marker under read heads
9	End of tape marker detected
10	Zero character (false end of block)
11	An instruction has been rejected as "done nothing", due to operator or hardware error.

If bit 2 is set to 1 the other bits of the control word have no meaning.

Bits 12-18 are undefined.

3) The error numbers are,

1	Error in opening file
2	Error in reading block
3	Error in writing block
4	Error in closing block
5	Incorrect handler number input
6	Handler not available
7	Incorrect file name (instead of 1)
8	Incorrect numbers input
9	No write permit fitted

Note the exception, error output:

\*CHN n m.

An invalid character has been input; n is the value of this character (in octal), m refers to the entry to subroutine QCHIN.

All programs halt after an error signal, re-entry is at 9.

### 1.5 Configuration

The minimum configuration for running these programs is a 903 central processor with 8192 words of store, teleprinter or paper tape input, a magnetic tape controller and at least one magnetic tape handler.

## Chapter 2 MTINIT

### 2.1 INTRODUCTION.

#### 2.1.1 Purpose.

MTINIT is designed to initialise a magnetic tape file by writing a header block on it in standard form. This header block will indicate either the name of the file to be written or that the tape is a work tape for general use. MTINIT must be used to prepare any tape before its first use by any other magnetic tape software.

#### 2.1.2 Form of Distribution.

The program is distributed as a sum checked binary tape for input by initial instructions. It may be convenient to dump the program onto magnetic tape using MSDUMP.

#### 2.1.3 Method of Use.

The number of the handler to be used in input and the program then lists the header block in the format as given for MTLIST, (Chapter 3), if there is a header block on the tape. The program then inputs the new file name and writes the new header block and an 'end of file' block. An alternate use is to input the handler number only to check to identity of a tape. This is the only routine which does not obey all file protection rules.

#### 2.1.4 Output.

A standard header block and 'end of file' block are written onto a magnetic tape.

### 2.2 FUNCTIONS.

#### 2.2.1 Entry Points.

Entry points are standard (see Section 2.7.9 paragraph 1.3.3) (i. e. location 32 for teleprinter, 33 for tape reader in the standard version)

#### 2.2.2 Operation.

On entry the program prints out the program name MTINIT as a heading. The operator then inputs the handler number which must be between 0 and 3 inclusive.

The program now attempts to read the first block of the tape. If this is a header block it will be listed in the format described in Chapter 3.2.3.1. If there is no header block a number of attempts will be made to read the first block. If the tape continues moving for more than 2 minutes (approx) RESET should be pressed and the program must be re-entered at 3; (35 in the standard version) If there is no header block on the file or when the header block has been listed the program will output:

INP NEW HDR

If paper tape input/output is being used the program will now halt and must be re-entered at location 9 to enable the operator to check any output before inputting a new header name. If the program is being used to check file contents or if the old file name has not been cleared for overwriting the program should be stopped at this point.

The operator may now input a new file name, the program will then use this file name to write a new header block and will then write an EOF block. When these blocks have been written the program will output:

OK

The program then halts, it may be re-entered at 9 to initialise another tape. No heading (MTINIT) will be output, the handler number required should be input immediately.

### 2.3 ERRORS.

Error messages are standard (see Chapter L.4) Re-entry after all errors is at 9, and the operator must re-start from inputting the handler number except for error CHN which will continue to read the next character.

### 2.4 STORE USED.

MTINIT	300 words approx.
Subroutines common to utility routines	470 words approx.
Magnetic Tape Routines	<u>770 words approx.</u>
TOTAL	1,540 words approx.

### 2.5 TIMING

The time taken by the program depends on types of tapes used and any previous information on them.



## Chapter 3 MTLIST.

### 3.1 INTRODUCTION.

#### 3.1.1 Purpose.

MTLIST will list the contents of specified blocks on a magnetic tape file according to instructions input to the program.

#### 3.1.2 Form of Distribution.

The program is distributed on a mnemonic tape for input by the SYMBOLIC INPUT ROUTINE (S. I. R.) The user is recommended to produce a version dumped on papertape or magnetic tape using T22 or MSDUMP. Tapes SSYS1 and MTBLOCK must also be assembled, followed by the last section of MTLIST.

#### 3.1.3 Method of Use.

The number of handler to be used and the file name are input and then program then lists the header block. The program then lists either the label blocks or any specified group of blocks according to the instructions input.

If a group of blocks are required the block numbers of the first and last of these blocks are input and they and all blocks between are listed, further lists may then be specified providing they do not require the tape to be rewound.

#### 3.1.4 Output.

The specified blocks are listed on the teleprinter.

### 3.2 FUNCTIONS.

#### 3.2.1 Entry Points.

Entry is standard (see Chapter 1.3.1)

#### 3.2.2 Operation.

On entry the program prints the program name MTLIST as a heading. The operator then enters the Handler number (between 0 and 3) and File name (up to 12 characters) and the program then opens the file and lists the contents of the header block (see 3.2.3.1).

To list label blocks the operator then enters L, and the program then lists all label blocks on the tape (see 3.2.3.2) and then rewinds and awaits input of further instructions.

To list any required group of blocks the operator inputs

B,

the number of the first block,  
the number of the last block,

and the program will list the required blocks, label blocks are always listed in the same format (see 3.2.3.2) the contents of other blocks are listed in octal (see 3.2.3.3). If only one block is required its number should be terminated by a colon (:) instead of a comma (,).

The program then awaits further instructions, unless an EOF block is detected when it will print

EOF and close the file, rewinding the tape.

To close the file the terminal ) is entered instead of L, or B, the program then prints

EOF and closes the file, rewinding the tape.

Example.

This is an example of a printout from MTLIST, only the instructions are printed in full.

MTLIST.

1, PROG FILE 1,

Header block is listed,

L,

Label blocks are listed

B, 2, 5:

Blocks 2 to 5 inclusive are listed

B, 17:

Block 17 is listed

) The file is closed and

EOF is printed.

3.2.3. Format.

3.2.3.1 Header Block

Location	Printed As
2; to 5;	Name of File, 12 alphanumeric characters.
6; to 8;	Integers
9; to 10;	Dates:Day Month Year as integers
11; to 17;	As integers

If 18; is zero no further information is printed, otherwise

18; to 27; Integers.

3.2.3.2 Label Block.

Location	Printed As
0;	BLOCK n where n is the block number.
2;	If positive (data label), printed as integer. If negative 2; and 3; are printed as alphanumeric label name, followed by the rest of the BLOCK as separate words printed in octal.
3; and 4;	If 2; was positive, 3; and 4; printed as 6 character alphanumeric data name.

Remainder of block.

If 2; contains EOF or EOY no further information is printed, and the tape is rewound.  
If 2; is #1, indicating a program dump remainder of block is printed as groups of 4 words, up to 20; or until the first word of a group is negative. The groups are printed as a pair of integers, followed by a pair of OCTAL Words.  
If any other type of label block, the remaining words are printed in octal, one word per line.

3.2.3.3 Data Blocks.

Location 0 is output as BLOCK n where n is the block number.

Locations 2; onwards are output in octal in groups of five numbers per line.

3.3 ERRORS.

Error messages are standard (see Chapter 1.4) and all re-entry is at location 9.

Error Numbers	Action on re-entry
1, 5, 6, 7, 9, 2,	Return to re-input handler number. Program will attempt to read next block on tape.
4, 8,	Program will attempt to close file. Re-input numbers of blocks requiring listing.
10,	(Wrong character input) Input L, B, or ) correctly.

3.4 STORE USED.

MTLIST.	420 locn. appx.
Subroutines common to utility routines	485 locn. appx.
Magnetic Tape Routines	<u>765 locn. appx.</u>
	1,670 locn. appx.

Note that QMWRITE may be omitted from the magnetic tape routines, it occupies 135 locns.

3.5 TIMING.

The time taken by the program depends on the types of tapes used and any previous information on them.

## Chapter 4 MSDUMP.

### 4.1 INTRODUCTION.

#### 4.1.1 Purpose.

MSDUMP is used to dump specified areas of store, containing program or data, onto a magnetic tape file, so that the information may be re-input to the same areas of store by the retrieval program MTCALL

#### 4.1.2 Form of Distribution.

The program is distributed as a mnemonic tape for input by the SYMBOLIC INPUT ROUTINE (S. I. R). The user is recommended to dump one or more versions onto paper tape or magnetic tape, using T22 or MSDUMP in another part of store. Tapes SSYS1 and MTBLOCK must also be assembled.

#### 4.1.3 Method of Use.

The handler number, file name and program name are input followed by pairs of addresses specifying the areas of store to be dumped onto tape.

After dumping each named program or data area either the file may be closed or further dumps specified.

The file to which the program is added must be already on the magnetic tape, and MSDUMP only opens the file for updating. (The header file must have been set up by MTINIT).

#### 4.1.4 Output.

A label block and the store blocks specified are written onto a magnetic tape.

### 4.2 FUNCTIONS.

#### 4.2.1 Entry Points.

Entry is standard (see Chapter 1.3.1).

#### 4.2.2 Operation.

On entry the program prints the program name MSDUMP as a heading. The operator then enters the Handler Number (between 0 and 3) and File Name (up to 12 characters)

the program then opens the file, (this may take several minutes if the tape has to be rewound.)

The operator then enters the

Program Name (up to 6 characters) and

Up to 4 pairs of addresses, specifying the areas to be dumped. Each address must be followed by a terminator (,) and the last address followed by a : (colon). When this program or data has been written onto the file correctly the program will output.

The operator may now either input ) which will cause the program to close the file, rewind the tape, and print EOR or input another program name and pairs of addresses. The program will always print EOR when the file is closed.

Example.

This is a sample of the print out to be expected from MSDUMP. The operator has updated the magnetic tape file PROGMAS<sup>T</sup>ER on handler 2 by adding two blocks of program SAMPL1 from location 32 to 117 and from location 250 to 600 and 1 block of program SAMPL2 from location 2048 to 3072 inclusive. He then closed the file and finished the run.

MSDUMP

2, PROGMAS<sup>T</sup>ER, SAMPL 1, 32, 117, 250, 600:

SAMPL2, 2048, 3072:

)

EOR

#### 4.3 ERRORS.

Error Mes sages are standard (see Chapter 1.4) and all re-entry is at location 9.

Error Numbers	Action on Re-entry.
1, 2,	Program will attempt to re-open file.
3,	No re-entry.
4,	Program will attempt to close file.
5, 6, 7	Input new handler number.
8,	(Incorrect addresses input) Re-input new addresses.

4. 4: STORE USED.

MSDUMP	340 locn. appx.
Subroutines common to utility routines	400 locn. appx.
Magnetic Tape Routines	<u>760 locn. appx.</u>
	1,500 locn. appx.

4. 5: TIMING.

The time taken by the program depends on the types of tape used and any previous information on them.

## Chapter 5 MTCALL

### 5.1 INTRODUCTION

#### 5.1.1 Purpose.

MTCALL is used to call down named programs from a magnetic tape file. The program must have been written onto the magnetic tape file by program MSDUMP.

#### 5.1.2 Form of Distribution.

The program is distributed as a mnemonic code tape for input by the SYMBOLIC INPUT ROUTINE (S.I.R.). Tapes SSYS1 and MTBLOCK must also be assembled, followed by the last section of MTCALL.

#### 5.1.3 Method of Use.

The handler number, file name and program name are input. The program then finds the program required on the file and places it in store after checking that it will not overwrite itself.

After calling a program either the file may be closed or further programs may be called.

#### 5.1.4 Output.

Only messages are output by this program.

### 5.2 FUNCTIONS.

#### 5.2.1 Entry Points.

Entry is standard (see Chapter 1.3.1)

#### 5.2.2 Operation.

On entry the program outputs the program name MTCALL as a heading. The operator enters the:  
Handler number (between 0 and 3) and  
File name (up to 12 characters)  
the program then opens the file. (This may take several minutes if the tape has to be rewound.)

The operator now enters the name of the program required, when this has been read correctly into store a ← is output.



The operator may either close the file and rewind by entering ) in which case the program will output EOR when the file has been closed, or enter another program name. The program always prints EOR when the file has been successfully closed.

### 5.3 ERRORS.

Error messages are standard (see Chapter 1.4) and all re-entry is to location 9.

Error Numbers	Action on re-entry.
1, 5, 6, 7, 11, 2,	Input new handler number. Program continues to search for label block of program.
4	Program attempts to close file.
10, (Program required overlaps MTCALL)	No re-entry, use MTCALL in different position.
11, (Program name not on specified file)	No re-entry at 9, check for wrong spelling or wrong file, re-enter at 0; or 1; if required.

### 5.4 STORE USED.

MTCALL	300 words approx.
Subroutines common to utility routines	400 words approx.
Magnetic Tape Routines	<u>760 words approx.</u>
TOTAL	1,460 words approx.

Note that QMWRITE may be omitted from the magnetic tape routines, it occupies 135 words.

### 5.5 TIMING.

The time taken by the program depends on the types of tape used and any previous information on them.