

ELLIOTT

9000

Volum. 4: ENGINEERING MAINTENANCE

Part 4: PERIPHERAL EQUIPMENT

Section 1: 9 Kc/s MAGNETIC TAPE (DPA341, DPA342, DPA343 & DPA344)

CONTENTS

As an interim measure this Section contains Appendices 1 to 4 inclusive. The remaining text will be published at an early date.

Appendix 1: 9 Kc/s MAGNETIC TAPE TEST PROGRAM, XTM70

Appendix 2: 9 Kc/s MAGNETIC TAPE TEST PROGRAM, XTM71

Appendix 3: 9 Kc/s MAGNETIC TAPE TEST PROGRAM, XTM72

Appendix 4: 9 Kc/s MAGNETIC TAPE TEST PROGRAM, XTM73

000E

Printed in England by
Engineering Unit, English Electric Computers Ltd.

Appendix 1: 9 Kc/s MAGNETIC TAPE TEST PROGRAM XTM70

CONTENTS

	Page
1. Purpose	1
2. Method	1
3. Operation	2
4. Sub-Routines	3
5. Bit Specification	9
6. Patterns Available	9
7. Meaningful Characters and Character Groups	9
8. Run-time Messages	10
9. Errors	10
10. Entry Points	11
11. Examples	11

Appendix 1: 9Kc/s MAGNETIC TAPE TEST PROGRAM XTM70

1. Purpose

To enable the user to construct simple programs for the testing of a 9 Kc/s Tape system, using macro - calls (subroutines) specified in section 4 below.

2. Method

2.1 Programs are constructed from a group of subroutines (or macro-programs) whose entry points are parameters set up by the assembler.

The subroutine names and parameters are given to the assembler by the operator.

The subroutine names may be:

- a) Typed in on the control teleprinter
or
- b) Punched on paper tape, and read in on the paper tape reader.

A line of code will be read into a buffer, and the required characters extracted and decoded.

If a syntax error occurs an error indication will be given. To run the sub-program the function GO must be typed.

2.2 Every function should be terminated by \textcircled{S} or $\textcircled{S}\textcircled{L}$.
Functions without variants may be terminated by a . or \textcircled{L} only.
e. g. JB. or IS..

2.3 Variants may be separated from the function by any number of spaces

e. g. WB F 256 . may be written as
 WB/F 256 .

3. Operation

3.1 The assembler program XTM70 will be supplied as a sumchecked binary (SCB) tape. The operator will set up the address 8180 (00000111 - - - 110101) on the word generator, load the tape into the reader, and press the "JUMP" button. The program will be read into store from location 21 upwards and will occupy approximately 5250 locations.

3.2 If it is desired to assemble from paper tape, load the "subroutine assembly" tape into the reader, and enter at location 23. The tape will be translated and the program will run when the function GO is identified.

3.3 If it is desired to type the subroutines on the control teleprinter, the entry point 23 should be used. The program will run when the function GO is typed.

3.2 should be entry
at 22 but doesn't
work properly.
Still reads from teletype
as for 23, but echos
input, so use SELECT
switch to override.

4. Subroutines

NAME	FUNCTION
OC	<p>Output Control</p> <p>May be followed by the required bit pattern If no bits are specified, the last pattern input under this function will be used.</p> <p>e. g. OC. Output Control OC n/ n2/ n3/ n4. Output Control bits n, n2, n3, and n4.</p>
WW	<p>Write-a-word</p> <p>May be followed by the required bit pattern, or if no bit pattern is specified, the last pattern specified under this function will be used</p> <p>e. g. WW. Write a Word WW n1/ n2/ n3/ Write bits n1, n2, and n3.</p>
WB	<p>Write-a-block</p> <p>Write a block of data to tape, using the block transfer facility. The data type may be specified, and also the block length.</p> <p>e. g. WB/A /256 Write a block of data Z of length 256 words F</p>
RW	<p>Read-a-word</p> <p>Reads one word from tape into store.</p>
RB	<p>Read-a-block</p> <p>Reads a block of data of the specified length.</p> <p>e. g. RB/512 Reads a block of length 512 words i. e. 1536 characters.</p>

4. Subroutines

NAME	FUNCTION
CB	Close-Block Terminates a write or read sequence.
IS	Input Status Input the status of the currently selected handler.
IW	Interrupt Wait: The function waits until an interrupt occurs from the system. When an interrupt occurs the program will exit from the dynamic loop, and continue obeying the subroutines. If an interrupt does not occur within 10 seconds the message WAITING will be displayed. This will be repeated every 10 seconds. Care should be exercised in using this function, as the time delay to obey the next subroutine may cause Missed Data Transfer.

The above functions are those used to define action to be initiated upon, or related to, the magnetic tape system. The maximum length of data transfer which may be specified should not exceed 512 words.

The functions given below are used to check and control the flow of the subroutine.

CP	Compare This must be followed by one of the following sets of variants. a) The pattern type, and the number of characters to be compared used in conjunction with the block transfer facility.
----	--

A
B
e.g. CP/F /n
Z

or

4. Subroutines.

NAME

FUNCTION

b) S and the number of bits it is required to compare with S refers to the last status word read, and the bits are those with which the status will be collated.

e. g. CP/S/n1/n2/n3/n4

or

c) D and the number of bits it is required to compare the data with. D is the last data word input under the read a word routine.

e. g. CP/D/n1/n2/n3/ .

PR

Print

The print function has the following variants:-

a) The variant M permits a message to be specified, which will be output at run-time. A string of characters may be typed after the variant M, and are terminated by a \$.

e. g. PR M THIS IS A MESSAGE \$.

To output a newline in a message string the character ↑ is typed. The characters (L) . () * may not be typed in a message string.

b) PR S. This prints out an 18 bit status word in the form:-

(L) ST 000000 000000 000000

c) PR D. This prints out the first data character found to be in error, as detected by the compare functions:-

A
CP/ B /n.
F
Z

(The data in the input area having been compared)
(with that transferred.)

4. Subroutines.

NAME

FUNCTION

If during the compare function no errors are found a marker is set. PR D will detect this marker and print the message

NO ERROR

If during the compare function an error occurred the display will be of the form:-

- Ⓛ WORD <N>
- Ⓛ EXPECTED <18 bit binary word>
- Ⓛ RECEIVED <18 bit binary word>

L

Label Declaration:

The user may declare labels at any point in the program. The labels must be of an integer form and are declared

L/n/ (some other function)

where n is the label number, and must be followed by some other function.

J

Jump:

The jump is unconditional and will jump up or down the program. The destination of the jump will be specified by an integer, under the label declaration.

e. g. J/n. where n is the label number.

JF

Jump If False:

This will jump to the specified label in the users program, if a flag called FALSE is set. This flag is not directly accessible to the user, but is set by the CP function if the compare functions have a zero result. The flag is examined by the JF function.

The jump is declared in the form

JF/n.

where n is the number of the label to which it is desired to jump.

4. Subroutines.

NAME	FUNCTION
LP	<p>Loop:</p> <p>This function enables the operator to set up loop counts. The counts may not be nested to a depth of greater than 16. The function is declared in the form:-</p> <p style="text-align: center;">LP/n.</p> <p>where n is the number of cycles to be made through the code which follows.</p>
JB	<p>Jump Back:</p> <p>This is used in conjunction with the LP function, and acts as the end of the section of code through which the loop is required. The LP and JB functions are inter-dependant and must be used together.</p> <p>N. B. These loops may be nested to a depth of 16 but may not intersect each other.</p>
CT	<p>Count:</p> <p>The function CT enables the operator to make use of the loop count facility LP in specifying a control word. When used in the following form:-</p> <p style="text-align: center;">LP 4. CT/OC (Loop dependant). (Note Code) OC (Loop dependant). JB OC (none loop dependant)</p> <p>The handler bits of the output control function will be modified dependant upon the two least significant bits of the count workspace</p>
WT	<p>Wait:</p> <p>The operator may specify the delay required, the number typed will be the delay in units of 100 usecs.</p>

4. Subroutines.

NAME

FUNCTION

N. B. due to the integer range of the machine the maximum delay will be in the region of 12 seconds.

Longer delays may be generated using the loop function round a wait function.

If no wait period is specified, the wait will become an operator wait, the message

WAITING

will be displayed every 10 seconds.

To continue after an operator wait entry point 9 should be used.

e. g. WT. Operator Wait, entry point 9 to continue.

or

WT 30000. Wait 3 seconds and continue.

NP

New Program:

This function cancels the current program and prepares to input a new program.

GO

Go

This terminates the sub-program areas, inserting a link to "assemble". (This prevents the sub-program from overrunning their bounds). A test is made to ensure that the LP and JB functions are fully specified and that all labels have been declared. If these requirements are satisfied the subroutines will be entered.

5. Bit Specification

To specify which bits of a word are to be set to the value 1, the operator will type a series of integers, with separators between each. Bit specification is used in functions OC, WW, CP, S, and CP D.

e. g. To set the following pattern 0001001101101101 the operator must type 1/3/4/6/7/9/10/13.

6. Patterns Available

Four standard patterns are available for the operator to use in block transfer.

They are as follows:-

<u>PATTERN</u>	<u>FUNCTION</u>
FULLHOUSE	F
RANDOM	B
ALL ZEROS	Z
ALTERNATE ONES AND ZEROS	A

For reference to these, see Section 4.

7. Meaningful Characters and Character Groups

Ⓢ : / Separators, may be used anywhere in the program.

. Ⓛ Function group terminators

§ Message string terminator

* Cancel this function

() Begin and end notes field

INTEGERS Where applicable

LETTERS Function names, variants, messages etc.

Ⓜ Stop assembly, wait for re-entry at 9.

↑ Newline in a "message" string only.

N. B. Except in message strings and notes field, alphanumeric groups are not permissible and will cause errors to be displayed.

8. Run-time Messages

At run-time the following messages, not specified by the operator, may be displayed:-

WAITING, displayed every 10 seconds when the program is either

- a) Waiting for an interrupt
- b) Waiting for operator re-entry.

N. B. It may be determined which function has caused this display by examining the interrupt level lamps. If the program is waiting for an interrupt, all lamps will be extinguished.

9. Errors

- 9.1 If an impermissible combination of characters are given, the message

SYNTAX

will be displayed. In teleprinter mode the line will be cancelled automatically, allowing the operator to re-type the line in error.

In paper tape mode the error is output on the punch, and compilation will continue, allowing the operator to examine all errors when compilation has finished.

- 9.2 If translation is complete, and an attempt is made to enter the subroutines, the dictionaries will be examined. If an incomplete LP/JB is found the message

Ⓛ INCOMPLETE LP/JB SEQUENCE

will be displayed. If a label is not declared the message

Ⓛ IDENTIFIER UNDECLARED <n>

will be displayed. If either of the above occur it will not be possible to run the program until either:-

- a) The program is corrected.
- or
- b) A 'dummy' label or sub-program entry is inserted,
where <n> is the label number.

9.3 Restrictions

Errors will occur if the following restrictions are not observed.

- 9.3.1. A P R M string should not contain any of the following characters * () . (L).
- 9.3.2. A notes field should not contain * (L), and should not be on a newline without a function.
- 9.3.3. Loops should not be nested deeper than 16. If this is not observed the error message
LP OVFLO
will be displayed.
- 9.3.4. No more than 50 labels should be declared. If this is not observed the message
DICT OVFLO
will be displayed.

10. Entry Points

The following entry points may be used by the operator.

- 21 Open multiplexer
- 23 Compile from Paper Tape
- 24 Entry to GO subroutine
- 25 Program debugging only.
- 9 Continue after HALT or WAIT

11. Examples

To write a 256 word block in odd parity (Random data) onto

handler 2

OC 2/6/9/10.	Control Word
WB B 256 .	Write Specification
CB .	Close Function

If it is desired to write say 5 blocks the above group may be inserted in a loop as follows.

LP 5 .	Setup loop count
OC 2/6/9/10 .	
WB B 256 .	
CB .	
JB .	Test and Jump

To obtain the status of handler 3 the operator should type the following instructions

OC 1/2	Select Handler 3
IS .	

If it is desired to print the status word the instruction PR S . should be added to the input instructions.

If a block of zero data has been written and read, it may be compared in store

- i. e. if the block is of 256 words the check would be
CP Z 256 .
JF 5 . (Jump to label 5 if comparison fails)

The individual status bits of the device can be examined by the CP S. Function.

- e. g. If it is desired to test if handler 2 is busy or in manual the following group of instructions may be used.
OC 2 . (select handler 2)
IS .
CP S 1/2 (examine bits 1 and 2)
JF 5 (go to 5 if set)

This permits branching of the program. If it is desired to read 5 blocks of data of length 512 words in odd parity and print a message if parity occurs or the device is put into manual, the following code would be input

LP 5 .
OC 2/6/8/10 .
RB 512 .
CB .
IS .
CP S 2 .
JF 5 .
CP S 4 .
JF 6 .
JB .
J 8 .
L/5/PR M ↑↑ Device in Manual \$
J 7 .
L/6/PR M ↑↑ Odd Parity \$
PR S .
L/7/J 7 .
L/8/PR M ↑↑ Reading Ok \$
J 7 .
GO .

Appendix 2: 9 K c/s MAGNETIC TAPE TEST PROGRAM XTM 71

CONTENTS

	Page
1. Purpose.. .. .	1
2. Test Tape	1
3. Control	1
4. Method	2
5. Error Messages	2
6. Tests	2

Appendix 2: 9 KC/S MAGNETIC TAPE TEST PROGRAM XTM 71

1. Purpose

To test the functions of the 9 Kc/s Magnetic Tape Controller attached to a 903 Central Processor with or without a Multiplexer.

2. Test Tape

The test tape is supplied as a sum-checked binary (SCB) tape to be input under initial instructions (i. e. entry point 8181).

3. Control

<u>Entry Point</u>	<u>Use</u>
9	Continue after wait or error
21	Open Multiplexer and stop
22	Test Handler 0,
23	Test Handler 1, no wait after error
24	Test Handler 2,
25	Test Handler 3,
26	Test Handler 0,
27	Test Handler 1, wait after error
28	Test Handler 2,
29	Test Handler 3,

To start the tests, the operator must:-

- (1) Position the tape as described in para. 4.
- (2) Open the Multiplexer, if fitted, by using entry point 21.
- (3) Enter the program at one of the entry points 22-29, as required.

4. Method

The operator positions the tape a few inches from the Beginning of Tape marker (B. O. T), switches the handler to Remote and ensures that the File Protect indicator on the handler is not lit. The operator then inputs the program.

5. Error Messages

In the following tests, the error messages are output either on the teleprinter, if available, or on the paper tape punch.

The messages are output in the form:-

TEST <M> ERR <N> <STATUS>

where M = number of test, N = error number, STATUS = Status Word of controller (three six-bit characters).

6. Tests

Position and State of Tape	Test No.	Test	Error Message
Few inches from B. O. T. with writing permitted, Remote.	1	The program sends a Rewind in Manual instruction and waits for 2 ms. The program then sends a Status Word instruction and checks that the Handler in Manual bit (bit 2) is set. If bit 2 is not set an error message is output. If bit 2 is set the message HANDLER TO AUTO is output. The operator puts the handler into Remote and re-enters the program at entry point 9. The program sends a Status Word instruction and tests that the handler is in Remote (i. e. Status bit 2=0).	ERR 1

Position and State of Tape	Test No.	Test	Error Message
	1 (cont.)	If bit 2 is still set an error message is output.	ERR 2
B. O. T., writing permitted, Remote.	2	<p>The program sends a Status Word instruction and tests the Write Permit bit (bit 7). If bit 7 is not set the program jumps to test 3, otherwise the message INHIBIT is output and the operator puts the handler into Local and removes the Write Permit ring.</p> <p>The operator then re-positions the tape at B. O. T., puts the handler into Remote and re-enters the program at entry point 9. The program sends a Status Word instruction and tests that the File Protect indicator on the handler is lit (i. e. Status bit 7=0). If writing is permitted an error message is output.</p>	ERR 1
B. O. T., writing inhibited, Remote.	3	<p>The program sends an Erase instruction with Interrupts Permitted. If an interrupt does not occur within 240 ms from sending the Erase instruction, an error message is output.</p> <p>The Status bit Done Nothing (bit 11) is tested by sending a Status Word instruction. If bit 11 is not set an error message is output.</p> <p>The program sends a Control Word to inhibit interrupts and to clear bit 11. A new Status Word instruction is then sent and if bit 11 is still set an error message is output.</p>	ERR 0
			ERR 1
			ERR 2

Position and State of Tape	Test No.	Test	Error Message
B. O. T. , writing inhibited, Remote.	4	The program sends a Prepare to Write instruction with Interrupts Permitted . If an interrupt is not received within 143 ms, an error message is output.	ERR 0
		The program then sends a Write a Block instruction. If an interrupt does not occur within 333 μ s from the end of the instruction, an error message is output.	ERR 1
		This is followed by a Close Block instruction. If the final interrupt is not received within 25 ms an error message is output.	ERR 2
		The Status bit Done Nothing (bit 11) is then tested. If bit 11 is not set an error message is output.	ERR 3
		Finally, the program sends an instruction to inhibit interrupts and to clear bit 11, which is then tested; if it is still set an error message is output.	ERR 4
B. O. T. , writing inhibited, Remote.	5	The Status Word instruction is sent and the Write Permit bit (bit 7) is tested. If bit 7 is set, the program jumps to test 7, otherwise the message PERMIT is output. The operator puts the handler to Local, replaces the Write Permit ring, puts the handler into Remote and re-enters the program at entry point 9. The program sends a Status Word instruction to test bit 7 and if the latter is still not set an error message is output.	ERR 1

Position and State of Tape	Test No.	Test	Error Message
B. O. T. , writing permitted, Remote.	6	<p>The program sends a Status Word instruction to test the Handler in Manual bit (bit 2). If the handler is in Remote (bit 2=0) the message HANDLER TO MANUAL is output. The operator puts the handler into Local and re-enters the program at entry point 9. Status bit 2 is tested again and if it is not set an error message is output.</p>	ERR 1
B. O. T. , writing permitted, Local.	7	<p>The Handler in Manual bit (bit 2) is tested and if it is set the message HANDLER TO AUTO is output. The operator puts the handler into Remote and re-enters the program at entry point 9. Bit 2 is re-tested and if it is still set an error message is output.</p>	ERR 1
B. O. T. , writing permitted, Remote.	8	<p>The program sends a Prepare to Write instruction with Interrupts Permitted. If an interrupt does not occur within 143 ms an error message is output.</p> <p>The program waits a further 325 μs and then sends a Write a Word instruction. If an interrupt is not received within 250 μs an error message is output.</p> <p>The Missed Data Transfer bit (bit 3) is tested and if it is not set an error message is output.</p>	ERR 0
Writing permitted, Remote-Write a 1 word block	9	<p>The program sends a Prepare to Write instruction with Interrupts Inhibited, and waits 14 ms. If an interrupt occurs during this time an error message is output.</p> <p>At the end of the 14 ms the program sends a Write a Word</p>	ERR 0

Position and State of Tape	Test No.	Test	Error Message
	9 (cont.)	instruction and if an interrupt is received within 250 μ s an error message is output. Then a Close Block instruction is sent and if an interrupt occurs within 25 ms an error message is output.	ERR 1 ERR 2
Erase 4 inches of tape	10	An Erase instruction with Interrupts Permitted is sent and an error message is output if an interrupt is not received within 111 ms.	ERR 0
Backspace over a 1 word block	11	A Backspace instruction with Interrupts Permitted is followed by a wait of 200 ms. If an interrupt does not occur during this time an error message is output.	ERR 0
Write a 1 word block	12	The program sends a Prepare to Write instruction with Interrupts Permitted. If an interrupt does not occur within 14 ms an error message is output. Then a Write a Word instruction is sent and an error message is output if an interrupt is not received within 250 μ s. Finally a Status Word instruction is sent and an error message is output if an interrupt is not generated. A further error message is output if the Missed Data Transfer bit (bit 3) is set.	ERR 0 ERR 1 ERR 2 ERR 3
Backspace over a 1 word block	15	A Backspace instruction with Interrupts Permitted is followed by a 33 ms wait. If an interrupt is not received during this time an error message is output.	ERR 0

Position and State of Tape	Test No.	Test	Error Message
Read a 1 word block	16	<p>The program sends a Prepare to Read instruction with Interrupts Permitted and outputs an error message if an interrupt is not received within 22 ms.</p>	ERR 0
		<p>Next the program sends a Read a Word instruction. If an interrupt does not occur within 250 μs an error message is output.</p>	ERR 1
		<p>This is followed by a Status Word instruction and if this instruction does not produce an answering interrupt an error message is output. The Missed Data Transfer bit (bit 3) is tested and if it is set an error message is output.</p>	ERR 2
Write a 5 word block	19	<p>A Prepare to Write instruction is followed by a 14 ms wait. An error message is output if an interrupt is not received during this time.</p>	ERR 0
		<p>The program then sends a Write a Block instruction to write 5 words on the tape. If an interrupt is not received within 333 μs from the end of the instruction, an error message is output.</p>	ERR 1
		<p>Finally, the program sends a Close Block instruction. If an interrupt does not occur within 21 ms an error message is output.</p>	ERR 2
Backspace over a 5 word block	22	<p>The program sends a Backspace instruction and outputs an error message if an interrupt is not received within 35 ms.</p>	ERR 0
Read a 5 word block	23	<p>A Prepare to Read instruction is sent by the program. If an interrupt does not occur within 22 ms an error message is output.</p>	ERR 0

Position and State of Tape	Test No.	Test	Error Message
	23 (cont.)	<p>Then the program sends a Read a Block instruction to read 5 words from the tape. An interrupt should be produced within 392 μs from the end of the instruction, otherwise an error message is output.</p> <p>The next instruction is Close Block, which should be answered by an interrupt within 11 ms. If it is not, an error message is output.</p> <p>A Status Word instruction is sent by the program to test bit 3 (Missed Data Transfer), bit 5 (Short Record) and bit 6 (Long Record). If any of these is set an error message is output.</p>	<p>ERR 1</p> <p>ERR 2</p> <p>ERR 3</p>
Backspace over a 5 word block	26	The program sends a Backspace instruction and if an interrupt does not occur within 35 ms an error message is output.	ERR 1
Read 6 words from a 5 word block	27	<p>A Prepare to Read instruction is sent by the program, which outputs an error message if an interrupt is not received within 22 ms.</p> <p>This is followed by a Read a Block instruction to read 6 words. If an interrupt does not occur within 392 μs from the end of the instruction, an error message is output.</p> <p>The program then sends a Status Word instruction and an error message is output if an interrupt is not generated.</p> <p>If bit 3 (Missed Data Transfer) is set an error message is output.</p> <p>If bit 5 (Short Record) is not set an error message is output.</p>	<p>ERR 0</p> <p>ERR 1</p> <p>ERR 2</p> <p>ERR 3</p> <p>ERR 4</p>

Position and State of Tape	Test No.	Test	Error Message
Backspace over a 5 word block	28	The program sends a Backspace instruction and if an interrupt does not occur within 35 ms an error message is output.	ERR 0
Read 4 words from a 5 word block	29	A Prepare to Read instruction is sent by the program, which outputs an error message if an interrupt is not received within 22 ms.	ERR 0
		This is followed by a Read a Block instruction to read 4 words. If an interrupt does not occur within 392 μ s from the end of the instruction, an error message is output.	ERR 1
		The program then sends a Status Word instruction and an error message is output if an interrupt is not generated.	ERR 2
		If bit 3 (Missed Data Transfer) is set an error message is output.	ERR 3
Write a 3 word block	30	If bit 6 (Long Record) is not set an error message is output.	ERR 4
		The program sends a Prepare to Write instruction. If an interrupt does not occur within 14 ms an error message is output.	ERR 0
		Then the program sends a Write a Block instruction to write three words, the middle word consisting of 3 zero characters. If an interrupt is not received within 333 μ s from the end of the instruction, an error message is output.	ERR 1
		Finally, the program sends a Status Word instruction, and an error message is output if an interrupt is not generated.	ERR 2
		ERR 3	

Only one zero character is written.

Position and State of Tape	Test No.	Test	Error Message
Backspace over a 3 word block	31	The program sends a Backspace instruction and if an interrupt is not generated within 35 ms an error message is output.	ERR 0
Read a 3 word block	32	A Prepare to Read instruction is sent by the program. If an interrupt does not occur within 22 ms an error message is output.	ERR 0
		Then the program sends a Read a Block instruction to read 3 words in odd parity. If an interrupt is not received within 392 μ s from the end of the instruction an error message is output.	ERR 1
		The next instruction is a Status Word instruction, which should be answered by an interrupt. If it is not, an error message is output.	ERR 2
		If bit 4 (Parity Error) is not set an error message is output.	ERR 3
		If bit 10 (Zero Character) is not set an error message is output.	ERR 4
Write a 5 word block	33	The program sends a Prepare to Write instruction. If an interrupt does not occur within 14 ms an error message is output.	ERR 0
		Next the program sends a Write a Block instruction to write 5 words. If an interrupt is not received within 333 μ s from the end of the instruction, an error message is output.	ERR 1
		A Status Word instruction is then sent, and an error message is output if an interrupt is not generated.	ERR 2
		If bit 3 (Missed Data Transfer) or bit 10 (Zero Character) is set, an error message is output.	ERR 3
		The program returns to test 8 and cycles tests 8 to 33 inclusive until Status bit 9 (End of Tape) is set, when the program jumps to test 34.	

Position and State of Tape	Test No.	Test	Error Message
Rewind	34	The program sends a Rewind instruction and then, after 1.1 ms, sends a Status Word instruction. If bit 1 (Handler Busy) is not set an error message is output.	ERR 1
	35	The program waits until the Handler Busy bit is reset and then tests that bit 8 (Load Point) is set. If it is not, an error message is output.	
	36	The program outputs the message END XTM 71 RESTART NO. OF CYCLES <P> where P is the number of cycles of tests 8 to 33 which had been completed when the end of tape was detected. The program then returns to test 8 and starts cycling again on the same handler. To test a different handler, the operator must press the Reset button and enter at a new entry point corresponding to the required handler (para. 3).	

Appendix 3: 9 K c/s MAGNETIC TAPE TEST PROGRAM XTM72

CONTENTS

	Page
1. Purpose..	1
2. Method	1
3. Operator Options	2
4. Patterns	3
5. Modes	3
6. Errors	4
7. Error Summary	6
8. Entry Points	8

Appendix 3: 9 Kc/s MAGNETIC TAPE TEST PROGRAM XTM72

1. Purpose

Provides a general engineering test and a mechanism test of the 9 Kc/s Magnetic Tape System.

2. Method

The program selects each available handler and performs the following operations:-

- (1) The program writes 16 blocks of data; the first block contains four words and each successive block is increased by 64 words until a block of 964 words is written.
- (2) The blocks written are read by backspacing once and reading, backspacing twice and reading, backspacing twice more and reading, and so on until all the blocks have been read. The blocks are then read in the forward direction until the last block has been read.
- (3) The program then returns and repeats operations (1) and (2). Initially the program performs these operation ten times on each handler, but an option is available whereby the operator specifies the number of cycles to be performed. If it is desired to continue to the End of Tape the operator must specify a large number (e.g. 2,000); the extra cycles will not be performed after the End of Tape is detected.

3. Operator Options

The program parameters, the purpose of the parameters and the initial values of the parameters are shown in the following list:-

Parameter	Purpose	Initial Value
Re-reads	Number of re-reads to be attempted on a block of data.	4
Re-write	Number of attempts to write a block of data.	3
Find	Number of attempts to locate a block.	6
Cycles	Number of cycles to be made of the write, back, read sequence.	10
Pattn	Number of pattern to be selected (see para.4.)	1

To change any parameter value, the operator must first select entry point 26, which causes the parameter names to be output. The operator must then type a new value, followed by a full-stop, after each parameter which is to be altered. Any parameter not requiring alteration must have an oblique stroke, (i.e. /) typed after it. If any character other than an oblique stroke, a numerical value or a full-stop is typed, the parameter name will be re-output.

NOTE: A full-stop must NOT be typed on its own.

4. Patterns

The following patterns may be requested by the operator as detailed in para.3.

<u>Pattern No. (typed by operator)</u>	<u>Selected Pattern</u>
1	Random
2	All ones
* 3	All zeros
4	Shifting ones
* 5	Binary count

* Zero characters should not be used in even parity.

Initially pattern No.1 is selected unless changed by the operator.

5. Modes

There are four modes of operation and an entry point is provided for each mode. The alternative modes are:-

- (1) Odd parity, block transfer - entry point 22
- (2) Odd parity, write/read a word - entry point 23
- (3) Even parity, block transfer - entry point 24
- (4) Even parity, write/read a word - entry point 25

The operator selects one of these to start the test.

6. Errors

There are three basic types of error output. These are:-

- (1) Output all errors - entry point 27.
- (2) Output all errors and wait re-entry at 9 - entry point 28.
- (3) Suppress all error output - entry point 29.

The program initially outputs all errors unless entry point 28 or 29 is used.

Error messages will be output only after the program has made a specified number of attempts to recover from the error. The initial values are shown in para. 3.

6.1 Writing Phase

The following errors may be output during the writing phase:-

WR ERR
HR <X> BLCK <Y>

and one or more of the following:-

PRTY <P>
MDT <M>
ZCH <Z>

where PRTY = Parity, MDT = Missed Data Transfer, ZCH = Zero Character.

6.2 Reading Phase

The following errors may be output during the reading phase:-

- (1) If the program is unable to locate the desired block and has failed after repeated attempts, the message

HR <X>
LST <X>, FND <Y>

is output.

- (2) If the block is located but the program is unable to read it without exceeding the specified number of attempts, the following message is output:-

RD ERR
HR <X> BLCK <Y>

and one or more of the following:-

PRTY <P>
MDT <M>
LRCD <L>
SRCD <S>
ZCH <Z>

where PRTY = Parity, MDT = Missed Data Transfer, LRCD = Long Record, SRCD = Short Record, ZCH = Zero Character.

- (3) When data has been read from the tape, either successfully or in error, it is examined in store and compared with the data which was written. If the data fails to compare, the following message is output:-

D ERR HR <X> BLCK <Y>
WD <Z>
WR <Expected word in binary>
RD <First word in error in binary>

6.3 Program Wait

If the program is in the 'wait after error' mode, the message

WAITING

is output before the program enters a dynamic stop.

The operator can select entry point 30, which dumps the pattern written and the pattern read on paper tape. When the dump is complete, the program enters a dynamic stop.

If the program fails to read a block of data after the specified number of attempts, the program outputs an error message and stops. If the operator wishes to make further attempts at reading this block, he should re-enter the program at entry point 31. The program then makes the specified number of attempts to read the block.

6.4 Abandoning a Handler

If, at any stage, a handler is put into Manual or is unavailable (e.g. writing is not permitted), the following message is output and the next handler is selected;

```
ABANDON HR <X>  
ST <Status word in binary>
```

XMT72 Issue 1
doesn't produce
a message

7. Error Summary

The operator may request an error summary by the following methods:-

- (1) If, during the running of the program, an error summary is required, a manual interrupt should be generated on level 3.
- (2) If a run has been abandoned, the error summary up to that stage of the program may be requested by the use of entry point 32.

The error summary will contain the following information:-

(1) Writing Phase

- (a) Number of re-writes.
- (b) Number of temporary errors in writing.
- (c) Number of permanent errors in writing.
- (d) Ratio of temporary errors/number of words.
- (e) Ratio of permanent errors/number of words.

(2) Reading Phase

- (a) Number of re-reads.
- (b) Number of temporary errors in reading.
- (c) Number of permanent errors in reading.
- (d) Ratio of temporary errors/number of words.
- (e) Ratio of permanent errors/number of words.

(3) Re-location

- (a) Number of temporary losses of position.
- (b) Number of attempts to locate a block.
- (c) Number of total losses of position.

7.1 Format

The error summary format is:-

```
ERR SUMMARY HR <X>
WR PHASE
REWRITES <n>
TEMP. ERR <m>
PERM. ERR <o>
TEMP. ERR RATIO <m> / <w> x 10000
PERM. ERR RATIO <o> / <w> x 10000
RD PHASE
REREADS <p>
TEMP. ERR <q>
PERM. ERR <r>
TEMP. ERR RATIO <q> / <w> x 10000
PERM. ERR RATIO <r> / <w> x 10000
LOSS OF POSITION
PERM. <s>
TEMP. <t>
RECOV. <u>
```

The error ratios in the Read and Write phases are cumulative; all other error counts are cleared after each error summary request.

The error summary is followed by the number of cycles completed on the handler, in the form:-

```
HR CYCLES = <n>
```

8. Entry Points

<u>Entry Point</u>	<u>Result of Entry</u>
21	Open Multiplexer and stop.
22	Odd parity, block transfer and start test

Entry Point	Result of Entry
23	Odd parity, read/write a word and start test
24	Even parity, block transfer and start test
25	Even parity, read/write a word and start test
26	Change parameters (see para. 3)
27	Set Output All Errors flag and stop
28	Set Wait After Errors flag and stop
29	Set Suppress All Errors flag and stop
30	Dump of written and read data areas
31	Restart reading block
32	Output error summary
9	Restart after an error wait.

Appendix 4: 9 K c/s MAGNETIC TAPE TEST PROGRAM XTM73

CONTENTS

	Page
1. Purpose..	1
2. Method	1
3. Errors	2
4. Error Recovery	5
5. Error Summary	5
6. Output	7
7. Entry Points	7
8. Selection of More than 1000 Blocks..	8

Appendix 4: 9 Kc/s MAGNETIC TAPE TEST PROGRAM XTM73

1. Purpose

To provide an interchangeability test for the 9 K c/s Magnetic Tape System, incorporating recovery routines, and requiring a minimum of operator intervention. The data written and read is random, in blocks of random length. The program incorporates comprehensive error counts which may be assessed by the user at any stage during the running of the program (see para.5).

2. Method

2.1 Writing Phase

Selecting each handler in turn, the program writes a number of blocks of random data. Initially 1000 blocks will be written, but the operator may request a greater number (see para.8).

The maximum permitted block length is 2048 words, the minimum permitted block length is 4 words (i.e. 1 word of data and a 3 word block header).

When writing is completed on each handler, the handler is rewound.

The program normally uses odd parity; a special entry point is provided to select even parity.

2.2 Reading Phase

When all handlers have been written, the data from each handler is read and checked in store. If any errors occur during reading or checking they are displayed on the control teleprinter (see para.3).

When the data has been read and checked on all the parent handlers (the handlers upon which the data was written), the program rewinds all handlers in manual and outputs the message

* * * CYCLIC TAPE CHANGE

When all tapes have been changed cyclically, the reading only phase is re-entered by the operator (entry point 30), and checking of the tapes continues on the new handlers.

The interchangeability test may be considered complete when all tapes have been read and checked on all handlers and have been returned to and checked on the parent handlers.

3. Errors

The program either waits after each error, or does not wait after error, depending on the entry point used.

It is only possible to dump the written and read data areas when the program is in the "wait after error" mode (see para.3.3).

If a control teleprinter is not fitted the errors are output on the paper tape punch, and may be printed off line.

3.1 Writing Phase

During the writing phase the following error message is output when the program fails to write a block of data after 12 attempts:-

WR ERR
HR <X> BLCK <Y>

and one or more of the following:-

PRTY <P>
MDT <M>
ZCH <Z>

where PRTY = Parity, MDT = Missed Data Transfer, ZCH = Zero Character.

The message

ABANDON

is output if the program is in the "wait after error" mode. The operator may then specify one of the recovery options (see para.4).

3.2 Reading Phase

If during the reading phase the program is unable to locate, after 6 attempts, the next block to be read, the message output is:-

HR <X>
LST BLCK <Y>
FND BLCK <Z>

where <Y> is the required block, and <Z> is the last block found in the search sequence.

The message

ABANDON

is then output if the program is in the "wait after error" mode. The operator may then specify one of the recovery options (see para.4).

If an error occurs after 4 attempts to read a block the following error message will be output:-

RD ERR
HR <X> BLCK <Y>

followed by one or more of the following:-

PRTY <P>
MDT <M>
LRCD <L>
SRCD <S>
ZCH <Z>

where PRTY = Parity, MDT = Missed Data Transfer, LRCD = Long Record, SRCD = Short Record, ZCH = Zero Character.

After data is read from the tape, either successfully or in error, it will be checked in store.

If a data error is found the following message will be output: -

D ERR HR <X> BLCK <Y>
WD <Z>
WR <18 bit binary word written>
RD <18 bit binary word read>

where the word read will be the first data word in error.

3.3 Dumping

If the program is in the "wait after error" mode, the operator may use entry point 31 to dump the data pattern written followed by a runout of blank tape, and the data pattern read. The processor will enter a dynamic stop and the operator may then select a re-entry point (see para.4).

3.4 Abandoning a Handler

If during any stage of the program a handler is put into Manual by the operator, or "Done Nothing" is set in the Status Word, or the handler becomes unavailable, the handler is abandoned.

No message is produced

When a handler is abandoned the message output is:-

ABANDON
<STATUS IN BINARY>

Any handler which is abandoned may not be tested again during the current running of the program.

4. Error Recovery

If an error occurs, or the handler is set to Manual by the operator, and it is desired to continue, one of the following recovery options may be used:-

- (1) Restart complete test
- (2) Restart handler phase)
- (3) Restart block phase) in either reading or writing phases
- (4) Abandon handler
- (5) Abandon block

An entry point is provided for each option (see para.7).

5. Error Summary

The operator may request an error summary by the following methods:-

- (1) If, during the running of the program, an error summary is required, a manual interrupt should be generated on level 3. To do this the operator must set the level 3 interrupt key in the MANUAL position, and then depress the Interrupt Level 3 button. The error summary will be output, and the program will then continue from the point where the interrupt occurred.

- (2) If a run has been abandoned, the error summary to that stage of the program may be requested by the use of entry point 25.

The error summary contains the following information:-

5.1 Writing Phase

- (1) Number of re-writes.
- (2) Number of temporary errors in writing.
- (3) Number of permanent errors in writing.
- (4) Ratio of temporary errors/number of words..
- (5) Ratio of permanent errors/number of words.

5.2 Reading Phase

- (1) Number of re-reads.
- (2) Number of temporary errors in reading.
- (3) Number of permanent errors in reading.
- (4) Ratio of temporary errors/number of words.
- (5) Ratio of permanent errors/number of words.

5.3 Relocation

- (1) Number of temporary losses of positions.
- (2) Number of attempts to locate the block.
- (3) Number of total losses of position.

6. Output

The error information is output in the form:-

```
ERR SUMMARY HR <x>
WR PHASE
REWRITES <n>
TEMP. ERR <m>
PERM. ERR <o>
TEMP. ERR RATIO <m>/<w> x 10 000
PERM. ERR RATIO <o>/<w> x 10 000

RD PHASE
REREADS <p>
TEMP. ERR <q>
PERM. ERR <r>
TEMP. ERR RATIO <q>/<w> x 10 000
PERM. ERR RATIO <r>/<w> x 10 000

LOSS OF POSITION
PERM. <s>
TEMP. <t>
RECOV. <u>
```

where <w> is the number of words written or read.

7. Entry Points

The following entry points are provided to enable the operator to start the test, and, if required, select the desired re-entry point:-

Entry Point	Purpose
21	Open Multiplexer
22	Interchangeability start test, wait after error
23	Start test, no wait after error
24	Input number of blocks required
25	Output error summary.
26	Restart handler phase
27	Restart block phase
28	Abandon handler
29	Abandon block
30	Restart reading phase
31	Dump written/read data
32	Select even parity

8. Selection of More Than 1000 Blocks

If it is required that more than 1000 blocks be written, the operator may use entry point 24. Use of this entry point outputs the message:-

NO. BLOCKS =

The operator then types the number of blocks required, which will always be greater than 1000. The number should be terminated by a full-stop. If an integer greater than 131071 is typed, an 'X' is output and the message phase repeated.