

Copy # 44

ENDICOTT PDL
October 7, 1958

MEMORANDUM TO:

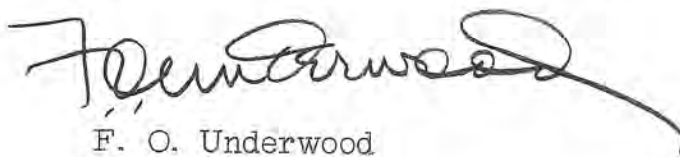
Mr. F. J. Furman

SUBJECT:

Expanded SPACE

The enclosed write-up defines the expansion of SPACE to include Calculation (3-address, Multiply, divide, half-adjust) High-Low-Equal compare, and Magnetic Tape attachment, essentially in final form.

Also included is a description of a possible Magnetic drum attachment for additional storage. The Drum attachment, as described, is the result of a brief look at such a possibility, no work being contemplated in the near future.



F. O. Underwood
Advisory Engineer
Dept. 280

FOU:mk
Encl.

October 1, 1958

FILE MEMORANDUM: SPACE
SUBJECT: SPACE Expansion

The purpose of this memo is to describe the Operation code structure for expanded SPACE, including calculation (MULTIPLY, DIVIDE, HALF ADJUST) and the attachment of Magnetic Tapes and a Magnetic Drum.

Three additional operation codes are required for calculation:

@ (A) (B) (C) - MULTIPLY

The multiplicand (factor A) multiplied by the multiplier (factor C) yields the product at address B.

This is a "reset" type of multiplication, that is, the product field is cleared at the beginning of the multiplication process.

There is no practical limit on the factor size. Both the A and C fields must be defined in length by word marks. The word mark in the product field is automatically set.

The multiplier will not be destroyed. Address 000 is used as a working area by the multiply routine, so that the programmer should not use this position for any data which must be held over multiplications.

% (A) (B) (C) - DIVIDE

The dividend (factor B), divided by the divisor (factor A) yields the quotient at address C.

This is, of course, a "reset" type of division, in that the quotient field will be reset, or cleared of any previous data.

There is no practical limit on the size of the factors involved.

Both the A and B fields must be defined in length by word marks. The word mark in the quotient field will be automatically set. The remainder will be left in the B field.

# (A) (B)	-	HALF ADJUST (2-address)
# (A) (B) (C)	-	HALF ADJUST (3-address)

The two-address form of this instruction causes a "5" to be added at location B, and the sign of field A to be moved one position to the left of location B. If there is a sign at location B, it is ignored when the "5" is added. Subsequent references to the half adjusted field must be to address "B-1".

The three-address form of the instruction causes the half correction (as above) and then "reset adds" the field (B-1) to address C.

If the A and C address parts of this instruction are equal, then the half-adjusted A field will effectively be shifted right, dropping the nonsignificant digits, so that subsequent references to this field may still be to address A.

Three-address Addition and Subtraction is permitted on the expanded SPACE machine.

+ (A) (B) (C)	ADD, $A + B + C = C'$
- (A) (B) (C)	SUBTRACT, $-A + B + C = C'$
R (A) (B) (C)	RESET ADD, $A + B = C$
S (A) (B) (C)	RESET SUBTRACT, $-A + B = C$

The conventional WORD MARK rules apply:

- 1) The C field must contain a word mark.
- 2) The A and B fields must contain word marks if they are shorter than the C field.

Three-address Editing is also permitted on the expanded SPACE machine.

E (A) (B) (C) EDIT, A (Mod)B = C

This form of the Edit instruction causes the data field at address (A) to be edited under control of the control word at (B) and the result to appear at address (C).

High-Low-Equal alphabetic comparing is a separate option in itself. The actual comparing of two fields (A) and (B) is effected as always with a "Compare" instruction:

C (A) (B)

Transfer of the program under the desired condition is controlled by the "Transfer Program" instruction:

T (I) T Transfer Program A>B
T (I) U Transfer Program A<B

The standard "088 Collator" sequence is used as a basis for High-low-equal comparison of Alphabetic numeric and special characters. The place of the "±" sign has not yet been fixed in this sequence.

In addition, the expanded SPACE machine provides for continual monitoring of the B field data on two-address instructions, and C field data on three-address instructions for "zero", "minus" and "overflow."

The "Transfer Program" instruction may be written as follows, in order to utilize this feature:

T (I) V Transfer Program on Zero

Following an "Add" or "Subtract" operation, for example, this instruction will cause a program transfer if the result was zero.

T (I) W Transfer Program on Minus

This instruction will cause a program transfer if the preceding (B) or (C) field was negative.

T (I) X

Transfer Program on Overflow

This instruction will cause a program transfer if the preceding arithmetic operation caused an overflow in the result field (either B or C).

The attachment of Magnetic Tape to the SPACE machine is a complete option in itself, independent of other options, and is accomplished by the inclusion of the Tape Adapter Unit (TAU).

The TAU device provides for the attachment of one or two Tape units of either 15 KC or 62.5 KC tapes. Only one tape may be controlled at a time. If only one Tape unit is attached, both reading and writing can be accomplished. If two Tape units are attached, then only one may be read, and the other only written. Selection of either tape for reading or writing is under control of a console switch.

Only one new Operation code is required for Tape operation:

U (d) TAPE CONTROL

The (d) character can be considered as an Operation modifier, the numeric part specifying the particular Tape operation to be performed, and the zone part specifying the particular tape unit. If the (d) is a 12-zone (A-B bit) character, the Tape Unit selected for reading is specified, and if the (d) is an 11-zone (B bit) character, the Tape Unit selected for writing is specified.

The numeric part of the (d) character specifies the following operations:

- | | |
|--------------------|--------------|
| 1. REWIND | (d = A or J) |
| 2. BACKSPACE | (d = B or K) |
| 3. ERASE FORWARD | (d = L) |
| 4. WRITE TAPE MARK | (d = M) |

In summary, the Tape Control instructions appear as follows:

UA	Rewind the "READ" Tape
UB	Backspace the "READ" Tape
UJ	Rewind the "WRITE" Tape
UK	Backspace the "WRITE" Tape
UL	Erase forward the "WRITE" Tape
UM	Place Tape Mark on the "WRITE" Tape.

Any of these instructions may be written as U (I)d so as to effect a Program Transfer.

In order to read and write Tape, the conventional "MOVE" and "LOAD" instructions are used, the address of the Tape being the 3-character code "TPE. "

M TPE (B) READ TAPE

This instruction will cause the Tape Unit selected for reading to start and the record to enter storage at the specified (B) address.

The (B) address must, of course, be the address of the high order position of the field in memory, since tapes are read and written in high-to-low order direction.

The tape will be read until an inter-record gap is detected, so that the programmer must provide an area in memory large enough to contain an entire group of records if grouped records are on the tape.

If word marks are in the memory area into which the tape is read, they will not be disturbed, nor will they affect the operation in any way.

M (A) TPE WRITE TAPE

This instruction will cause the Tape Unit selected for writing to start, and the contents of memory, starting at address (A) to be written on the tape. The operation will end when a group mark is detected in storage.

Word marks in the memory will not be disturbed, nor will they affect the operation.

L (A) TPE LOAD TO TAPE

This instruction will cause the Tape Unit selected for writing to start, and the contents of memory, starting at address (A), to be written on the Tape, until a group mark is detected in storage.

All word marks in the A field will cause an extra character to be written on the tape. This extra character will be called a "word separator. "

This word separator will be written for any position containing a word mark before the character in that position is written.

L TPE (B) LOAD FROM TAPE

This instruction causes the Tape Unit selected for reading to start, and the record to enter storage until a inter-record gap is detected. Any "word separator" read from the tape will be converted to a word mark, and the character following will also be written into the same position of memory as the word mark.

Tape operation requires an additional use of the standard "Program Transfer" operation code to permit branching on a Tape read or write error:

T (I) Y PROGRAM TRANSFER ON TAPE ERROR

The execution of this instruction will cause the Tape Error Indicator to turn off.

T (I) Z TRANSFER PROGRAM ON TAPE MARK

The "Tape Mark" will be written on Tape as a single character record, and indicates the "End of file" condition.

When this "tape mark" has been read, an "End of Tape File" latch is set.

The T (I) Y instruction then causes a program skip to (I) and resets the "End of Tape File" latch.

When grouping records on tape, the usual procedure is to separate the records by a "record mark". The record mark is the same as the plus (+) sign, code A-8-2.

Therefore, in order to insert record marks in memory for data to be written on tape, a constant (+) is required in storage, and is moved to the proper locations by the usual "MOVE" instruction.

The handling of the Group Mark is essentially the same, except that the code for this character is A-B-8-4-1. It can be entered into memory as a constant during the program loading operation by punching the card with an E and an 8 under-punch. This character cannot be printed.

The following controls are added to the Console for Tape operation:

- 1) TAPE STATUS SWITCH
Position 1: TU #1-READ
 TU #2-WRITE
Position 2: TU #1-WRITE
 TU #2-READ
- 2) BACKSPACE KEY, TU #1
- 3) BACKSPACE KEY, TU #2
- 4) TAPE ERROR INDICATOR LIGHT, TU #1
- 5) TAPE ERROR INDICATOR LIGHT, TU #2
- 6) END OF FILE LIGHT

The attachment of Magnetic Tape to the SPACE machine yields a system with the following primary functions:

- 1) Magnetic Tape to Printer.
- 2) Magnetic Tape to Cards
- 3) Punched cards to Magnetic Tape.

Combinations of these functions permits the following operations:

- 1) Searching of tapes under card control with selected records extracted and transcribed to documents, cards or an output tape.
- 2) Updating of tape records from transaction cards, simultaneously producing a report and/or summary cards.

- 3) Tape-to-tape operations involving data editing and grouping of records, or merging card transactions with a tape file, preliminary to processing on a larger system.
- 4) Merging of Tape (master file) data with Card (transaction) data for the production of bills, invoices, etc., on the SPACE machine.
- 5) Card to tape conversion with data auditing and record grouping. Transaction registers and/or exception cards may be produced as by-products.

A magnetic drum may also be attached to the SPACE machine as a separate option. No additional Operation codes are required.

The drum will be mechanically the same as the 650 drum, but with improved bit density and increased capacity, fully alphanumeric.

The full capacity of the drum will be 60,000 characters, arranged into 600 records of 100 characters each, and may be obtained in increments, as follows:

- 1st option, 15,000 char. total
- 2nd option, 30,000 char. total
- 3rd option, 60,000 char. total

Transfer of data from drum-to-core and core-to-drum is effected by conventional "MOVE" and "LOAD" codes, but with a "modifier" or (d) character. The "d" character specifies whether the (A) or (B) part of the "MOVE" or "LOAD" instruction is a drum address:

M (A) (B) A MOVE, DRUM-TO-CORE

In this instruction, the (d) digit is an "A", meaning that the (A) address refers to one of the 600 drum addresses, and (B) is a core address. One one-hundred character record will be moved from the drum to the cores, high-order digit first. Word marks in the core area will not be disturbed, nor affect the operation.

M (A) (B) B MOVE, CORE-TO-DRUM

This instruction is similar to the above, except that the 100-character record in core storage, whose high-order character is at address (A), will be moved to drum address (B).

Again, word marks in core storage are not disturbed, nor do they affect the operation.

L (A) (B) A LOAD, DRUM-TO-CORE

This instruction will cause the 100-character record at drum address (A) to load core storage, starting at address B. Any word marks in this area of core storage will be automatically removed. Any "word-separator" characters will cause a word mark to be written in the corresponding location in core storage, and the next character from the drum will be written into the same location.

L (A) (B) B LOAD, CORE-TO-DRUM

This instruction will cause an area of storage (high-order character at location A) to load to the drum, starting at drum address (B).

The operation terminates upon the detection of a "GROUP MARK" in memory, just as with tape operation.

Each word mark in the core storage area is converted to a "word separator" character, written on the drum, and the character in the same core storage location will be written in the following position of the drum.

If the record from core storage is of such a nature that it causes more than 100 characters to be written on the drum, automatic record overflow will occur, until the "Group Mark" is detected.

The drum option provides for the use of an additional (d) character for the T (I)d instruction, to permit "Program Transfer on Drum Error. "

Following the reading or writing of a drum record, an instruction,

T (I)N PROGRAM TRANSFER, DRUM ERROR

permits the re-reading or re-writing of the record in the event that a single-bit error should occur.

As mentioned above, the drum will be mechanically the same as the 650 drum, but with increased capacity. The access time remains at a maximum of 4.8 ms. Characters are transferred to and from the drum at a rate of 16 us per character, 1.6 ms per 100-character record.

The use of the "MOVE" and "LOAD" instructions permit the use of the drum for storing data in record or table form, and also for the storage of programs.

Appended to this memo are two charts, one showing the latest Operation Code structure, and the other showing the assignment of the (d) character for the T(I) d instruction.

Note that all of the codes of the optional features are included, and that no codes are shown for "float dollar" and "fill asterisk", as these operations are now specified by the edit control word.

Also appended to this memo are two "Output Speed" graphs, showing the output speeds for both 15 KC and 62.5 KC magnetic tape, on a "Tape-to-printer" operation.

It can be seen that for the majority of applications, where the records are from 200-600 characters in length, and correspondingly, one to five lines are printed per record, that the output speed is generally in the range of 550 lines per minute.

Other tape operations do not lend themselves well to such a graphical interpretation because of the synchronous nature of the card feed and card punch mechanism.

However, it is practical to write ungrouped 80-column records from cards at a rate of 800 cards per minute, if the processing time per record does not exceed 14.6 ms.

Grouped records may be written at high rates, and more processing time per record is available. For instance, allowing the full 21 ms processing time per record, 4 records per group are obtained at $4/5 (800) = 640$ cards/min., and 10 records per group are obtained at $10/11 (800) = 727$ cards/min.

Tape-to-card operations may proceed at a rate of 250 CPM within very practical limits of processing time and tape record size.

FOU:mk

SPACE OPERATION CODES

NUM.	NO - ZONE		12 - ZONE		11 - ZONE		0 - ZONE	
b1	b1		&		-	SUBTRACT		
1	1	FEED	A	CARR. SKIP AFTER	J		/	CLEAR
2	2	PRINT	B	CARR. SKIP BEFORE	K	STACKER SELECT	S	RESET SUBTRACT
3	3	PRINT-FEED	C	COMPARE	L	LOAD	T	TRANSFER PROGRAM
4	4	PUNCH	D	MOVE DIGIT	M	MOVE	U	TAPE UNIT CONTROL [‡]
5	5	FEED - PUNCH	E	EDIT	N	NO OP.	V	
6	6	PRINT - PUNCH	F		O		W	
7	7	PRINT-FEED-PUNCH	G		P	CARR. SPACE BEFORE	X	
8	8	FEED RELEASE [‡]	H		Q	CARR. SPACE AFTER	Y	MOVE ZONE
9	9	PUNCH RELEASE [‡]	I		R	RESET ADD	Z	MOVE-ZERO SUP.
8-2	0						+	ADD
8-3	#	1/2 ADJUST [‡]	.	STOP	\$,	SET WORD MARK
8-4	@	MULTIPLY [‡]	◇	CLEAR WORD MARK	*		%	DIVIDE [‡]

[‡] OPTIONS

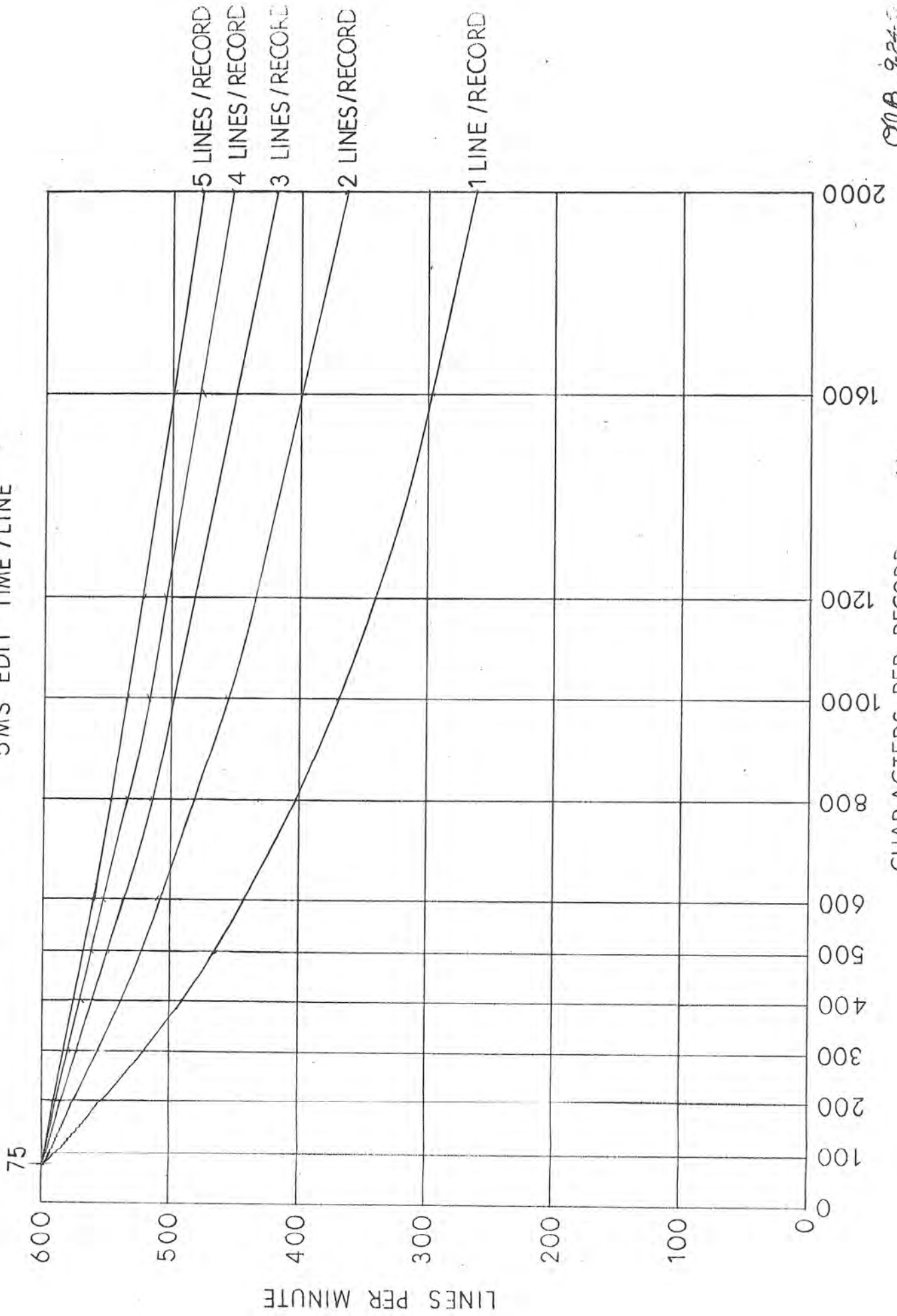
3 OCT. 58 JAW

SPACE d-CODE FOR T(I, J), TRANSFER PROGRAM INSTRUCTION

NUM.	NO-ZONE		12-ZONE				11-ZONE		0-ZONE																									
	b1	b2	&	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	+	,	%		
1	1	UNCONDITIONAL	#	A	'LAST CARD SWITCH ON'	"	'C' "	#	J	SENSE SWITCH 'J' ON #	/	UNEQUAL COMPARE																						
2	2	"	#	B	SENSE SWITCH 'B' ON #	"	'D' "	#	K		S	EQUAL COMPARE																						
3	3	"	#	C	"	'E' "	#	L	TAPE ERROR	#	T	A > B																						
4	4	"	#	D	"	'F' "	#	M	TAPE MARK	#	U	A < B																						
5	5	"	#	E	"	'G' "	#	N	DRUM ERROR	#	V	ZERO																						
6	6	"	#	F	"	'H' "	#	O			W	OVERFLOW																						
7	7	"	#	G	"	'I' "	#	P			X	MINUS																						
8	8	"	#	H	"			Q			Y																							
9	9	"	#	I	"			R			Z																							
8-2	0	"	#	10	#						+																							
8-3	#	"	#	11	#	.					,																							
8-4	@	"	#	12	#	◇					%																							

OPTIONS -

SPACE OUTPUT SPEEDS—15 KC TAPE TO PRINTER
 BASED ON 10 MS TAPE START TIME
 5MS EDIT TIME/LINE



9245

SPACE

OUTPUT SPEEDS — 62.5 KC TAPE TO PRINTER
BASED ON 10 MS TAPE START TIME
5 MS EDIT TIME / LINE

