



**Information  
Systems**

**World Leader  
In Time-Sharing  
Service**

**Vocabulary  
for  
Mark I  
Time-Sharing  
Service**

*4800 Core word  
20 Bit 40 word  
2 word  
Forthin 5300 word*

**GENERAL  ELECTRIC**



## SYSTEM COMMANDS

### DIRECTIVE COMMANDS

<b>CONTROL; SHIFT; @</b>	Stops program execution and type-outs. (Pressed simultaneously)
<b>RETURN</b>	Terminates a program line, causes the system to take action based upon input provided, and acts as a normal carriage return.
<b>ALT MODE</b> ● or <b>ESCAPE</b> ● or <b>CONTROL</b> ● with X	Deletes an input line as if nothing had been typed.
<b>ARROW</b> (←) ● or <b>UNDERLINE</b> (—) ● (with Shift Key)	Erases the last character(s) typed, one for each arrow or underline.
<b>BYE</b> or <b>GOODBYE</b>	Disconnects terminal from the system.
<b>HELLO</b>	Allows changes in user number without disconnecting terminal from the system.
<b>NEW</b>	Introduces a new program.
<b>OLD</b>	Retrieves from storage a previously saved program.
<b>RENAME</b>	Changes the name of a current program.
<b>RUN</b>	Compiles and executes the current program.
<b>SAVE</b>	Stores, for future use, the current program under the current name.
<b>SCRATCH</b>	Eliminates the working copy of a program. (The name of the program is retained.)
<b>STOP</b>	Causes the system to disregard all previous commands not yet executed (except when a type-out is occurring) and stop.
<b>SYSTEM</b>	Changes the compiler.
<b>UNSAVE</b>	Removes a previously saved program entry from the reference catalog, but does not remove program from working area.

● Special key on terminal unit.

### EDIT COMMANDS

<b>EDIT DELETE</b>	Removes portions of a program.
<b>EDIT DUPLICATE</b>	Duplicates program lines.
<b>EDIT EXTRACT</b>	Retains portions of a program.
<b>EDIT LIST</b>	Provides a listing of specified portions of the current program. If portions are not specified, the listing proceeds from the last statement to the first.
<b>EDIT MERGE</b>	Adds several saved programs together with new line numbers to form a single current but not saved program.
<b>EDIT MOVE</b>	Moves a program section to another portion of the program.
<b>EDIT PAGE</b>	Lists up to 9 programs on sequentially numbered pages.
<b>EDIT RESEQUENCE</b>	Renumbers the lines of the current program.
<b>EDIT RUNOFF</b>	Provides an edited version of saved programs (indentation, centering, justified margins, line spacing, etc.)

## SYSTEM COMMANDS (Continued)

<b>EDIT TEXT</b>	Lists up to 9 programs on sequentially numbered pages, but with no line numbers.
<b>EDIT WEAVE</b>	Combines several saved programs according to existing line numbers to form a single current but not saved program.

In addition, 20 comprehensive functions are available for string manipulation.

### INFORMATIVE COMMANDS

<b>CASALOG</b>	Lists the names and lengths of all programs in the user's catalog.
<b>CATALOG</b>	Lists only the names (not lengths) of all programs in the user's catalog.
<b>LENGTH</b>	Gives the approximate length in character count of the current program.
<b>LIST</b>	Lists the entire current program with a heading.
<b>LIST-n</b>	Lists, with a heading, the current program beginning with line number n.
<b>LISTNH</b>	Lists the entire current program with no heading.
<b>LISTNHn</b>	Lists, with no heading, the current program beginning with line number n.
<b>STATUS</b>	Gives present status of the current program.
<b>TTY</b>	Gives the present status of the terminal, including the DATANET*-30 channel number, current user number, program name, system name, and program status.

\*DATANET is a Reg. Trademark of the General Electric Company.

### MODE

<b>DSM</b>	Enters data storage mode (input).
<b>KEY</b>	Enters keyboard mode (input).
<b>PLOT</b>	Enters plot mode (output).
<b>TAPE</b>	Enters paper tape mode (input).

## BASIC LANGUAGE

### OPERATIONS

<b>+ Plus</b>	<b>/ Divide</b>
<b>- Minus</b>	<b>↑ To the power (underline (—) on some terminals)</b>
<b>* Multiply</b>	<b>() Parentheses</b>

Note: No implied operations

### FUNCTIONS

<b>SIN (X)</b>	Sine of X	} X must be in radians
<b>COS (X)</b>	Cosine of X	
<b>TAN (X)</b>	Tangent of X	
<b>ATN (X)</b>	Arctangent of X	
<b>EXP (X)</b>	Natural exponential of X, e <sup>X</sup>	
<b>ABS (X)</b>	Absolute value of X  X	
<b>LOG (X)</b>	Natural logarithm of  X	
<b>SQR (X)</b>	Square root of  X	
<b>RND (X)</b>	Generates a random number between 0-1 at each call	
<b>INT (X)</b>	Integer part of X	



## BASIC LANGUAGE (Continued)

### ARITHMETIC STATEMENTS

<b>DEF</b>	DEF FN <b>letter</b> (unsubscripted variable)=expression 5 DEF FNG(Z) = 1 + Z*Z)
<b>DIM</b>	DIM <b>letter</b> (integer) or DIM <b>letter</b> (integer, integer) 5 DIM A(17) or 10 DIM B(3, 20)
<b>LET</b>	LET <b>variable</b> = <b>expression</b> 5 LET X1 = Y + Z + (Z/A - B↑D1)
<b>Multiple Variable Replacement</b>	LET <b>variable</b> = <b>variable</b> = <b>expression</b> 5 LET A = B = 10
<b>String Variable</b>	LET <b>string variable</b> = <b>string variable</b> or <b>string</b> (15 characters of alphanumeric data) 5 LET A\$ = "JULY" 5 LET A\$(1) = B\$(20)

Note: Each Statement must have a line number.

### IDENTIFIER STATEMENTS

<b>DATA</b>	DATA <b>number, number, . . . , number, string, string</b> 5 DATA 1, 2, -3, 7, 123.479, -2.35E-4, AUG, 912, . . .
<b>END</b>	5 END
<b>REM</b>	REM <b>any string of characters whatsoever</b> 5 REM THIS IS THE END OF APPENDIX C

### INPUT/OUTPUT STATEMENTS

<b>INPUT</b>	INPUT <b>variable, variable, . . . , variable</b> 5 INPUT X, Y, Z, A1, Q(I, J), C\$
<b>PRINT</b>	PRINT <b>label</b> or PRINT <b>label expression</b> or PRINT <b>expression</b> 5 PRINT "SINE" or 5 PRINT "X="; X(I, K) or 5 PRINT A + B*COS(Y)
<b>TAB Function</b>	TAB (terminal carriage placement) 5 PRINT TAB (31)
<b>READ</b>	READ <b>variable, variable, . . . , variable</b> (must be accompanied by DATA statement) 10 READ X, Y, Z, A1, Q(I, J), A\$, B\$(2)
<b>RESTORE</b>	10 RESTORE (allows data to be reread)

### LOGIC STATEMENTS

<b>GOTO</b>	GOTO <b>line number</b> 5 GOTO 17
<b>Computed GOTO</b>	On (Expression) GOTO (Line No.), (Line No.), . . . 5 ON X + Y GOTO 10, 20, 30, 40 . . . 10 - IF INT(X + Y) = 1 20 - IF INT(X + Y) = 2 . .
<b>IF-THEN</b>	IF <b>expression relation expression</b> THEN <b>line number</b> 5 IF X + Y > 0 THEN 419
<b>STOP</b>	10 STOP

## BASIC LANGUAGE (Continued)

### LOOP AND SUBROUTINE STATEMENTS

<b>FOR</b>	FOR <b>unsubscripted variable</b> = <b>expression</b> TO <b>expression</b> STEP <b>expression</b> 5 FOR X1 = 0 TO 7 STEP 0.5 or 5 FOR I = 1 TO 17 (STEP of 1 assumed)
<b>GOSUB</b>	GOSUB <b>line number</b> 10 GOSUB 110
<b>NEXT</b>	NEXT <b>unsubscripted variable</b> 10 NEXT X
<b>RETURN</b>	10 RETURN
<b>CALL</b>	CALL (Subroutine Name) 5 CALL SUBR
<b>CHAIN</b>	CHAIN (Saved Program Name) or CHAIN (Saved Program Name), Line Number 5 CHAIN PROG 5 CHAIN PROG, 510

### MATRIX STATEMENTS

<b>Matrix Commands</b>	1. Must be doubly dimensioned .. A(1, 3) 2. Rows and columns are subscripted from 1 to n 3. Redimensionable to larger or smaller matrix
<b>MAT READ letter</b> (expression, expression)	MAT READ A(M, N)
<b>MAT PRINT letter</b>	MAT PRINT A
<b>MAT letter = matrix expression</b>	MAT C = A + B MAT C = A - B MAT C = A * B
<b>MAT letter = (expression) * letter</b>	MAT C = (k) * A
<b>MAT letter = ZER</b> (expression, expression)	MAT C = ZER(M, N)
<b>MAT letter = CON</b> (expression, expression)	MAT C = CON(M, N)
<b>MAT letter = IDN</b>	MAT C = IDN
<b>MAT letter = IDN</b> (expression, (same) expression)	MAT C = IDN(M, M)
<b>MAT letter = TRN</b> (letter)	MAT C = TRN(A)
<b>MAT letter = INV</b> (letter)	MAT C = INV(A)

### FILE COMMANDS

<b>FILES</b>	Saved file designation, must be first statement 5 FILES A;B;C
<b>SCRATCH #</b>	SCRATCH # (File No. as specified in file statement) 5 SCRATCH #1 (Initiates Write Mode)
<b>RESTORE #</b>	RESTORE # (File No.) 5 RESTORE #1 (Initiates Read Mode)
<b>READ #</b>	READ # (File No.), A, B 5 READ #1, A (Read #1 into A)
<b>WRITE #</b>	WRITE # (File No.), A; B 5 WRITE #1, A (Write A into #1)
<b>BACKSPACE #</b>	BACKSPACE # (File No.), Backspace one data point 5 BACKSPACE #1



**STATEMENTS**

<b>ARITHMETIC</b>	100 A=B-1.; C=D=E=5.; GC=AS/1+2/TLA 110 F(T+B/3) = FR + (FI=FS*GS/3)*COS (FI) 120 ML=MAN - "LIC"
<b>ARITHMETIC STATEMENT FUNCTION</b>	100 SINH (X) = .5*(EXP(X)-EXP (-X) )
<b>INTERNAL FUNCTION</b>	100 RUST (X): ... 110 KANS (A, B): ... 120 REAL, JUST ( ): ... 130 INTEGER, FIRST(L): ...
<b>ASSIGN ... TO ...</b>	100 ASSIGN 6 TO J 110 ASSIGN FORMAL, TO R 120 ASSIGN A5, TO A6
<b>BACKSPACE</b>	100 BACKSPACE 110 BACKSPACE 3 120 BACKSPACE I
<b>CALL</b>	100 CALL FEN (S, A(5), B(L/2) ) 110 CALL OUT 120 CALL B(N)
<b>COMMON</b>	100 COMMON A(12), B, C(3,2), K, J(4,3,2,2) 110 INTEGER COMMON SUM, LINE, BSL(15)
<b>CONTINUE</b>	100 25 CONTINUE (The word CONTINUE may be omitted because empty statements may be labelled.) 110 NEXT: CONTINUE 120 NEXT:
<b>\$DATA</b>	100 \$DATA 110 \$DATA PAYNOS, CASES (where PAYNOS and CASES are saved DATA files)
<b>DIMENSION</b>	100 DIMENSION A(5), LOST(45, 12)
<b>DO</b>	100 DO 16 I = 1, 10 110 DO ALL, I = K, J, 2 120 DO 25, X = 3, 17., .5
<b>END</b>	100 END (Not needed except before main program.)
<b>END internal</b>	100 END INTERNAL 110 END RUST
<b>ENDFILE</b>	100 ENDFILE 110 ENDFILE 5 120 ENDFILE IUNIT
<b>ENTRY</b>	100 ENTRY BACKALWAYS (TO, FROM) 110 ENTRY AFT1(X) 120 ENTRY SUMPKB 130 ENTRY REPEAT(N)
<b>EQUIVALENCE</b>	100 EQUIVALENCE (BEGIN, START, INITIATE), 105 + (D(3), B(5), L(2), K(5) ) 110 INTEGER EQUIVALENCE (TFG, TFD, TFR)
<b>EXTERNAL</b>	100 EXTERNAL HUNCH, DRAG
<b>\$FILE</b>	110 \$FILE MP, MCOST, VENDOR 120 \$FILE INV1/INV2/INV3/INV4, SUM
<b>FORMAT</b>	100 LINE:FORMAT (I4, 4E12.5) 110 77 FORMAT ("NO. OF CASES", I2, 3A3)
<b>FUNCTION</b>	100 FUNCTION AFT 110 INTEGER FUNCTION HUNCH (L, T)
<b>GOTO</b>	100 GOTO 13 110 GOTO EXTRA
<b>GOTO (...)</b>	100 GOTO (12, LAST, KONLY, 15) AFTER 110 GOTO (M1, M2, MT3), M

<b>IF (...)</b>	100 IF(A) 25, 26, 27 110 IF(A*SIN(B) ) AGAIN, EXCEPT 120 IF(J=K/3) 3 130 IF(IFR-"YES") TRUST, UNT
<b>IF (ENDFILE)</b>	100 IF(ENDFILE 3) 45, EOF 110 IF(ENDFILE J) MORE
<b>INPUT (terminal)</b>	100 INPUT REPLY, QUAN 110 INPUT, (COEF (I), I = 2, 10)
<b>INTEGER</b>	100 INTEGER J(3), TRUD(12,2,3) 110 INTEGER A, S, TLA(16), TLB(16)
<b>\$OPT</b>	100 \$OPT SS 110 \$OPT SIZE 120 \$OPT REAL 140 \$OPT IFF BOTH
<b>PAUSE</b>	100 PAUSE 110 PAUSE SENSESWITCH3 120 PAUSE "YES"
<b>PRINT (terminal)</b>	100 PRINT 45, A 110 PRINT REP, (A (I), I=1, 10) 120 PRINT, A, A*B, A/BCA+3.2 130 PRINT "MORE OR LESS", ↑TABLE, T(I+2)*B
<b>READ (...)</b> (temporary file)	100 READ, A, F, G 110 READ TITLE 120 READ 12, F, SYT
<b>READ (...)</b> (permanent file)	100 READ (3, TITLE) 110 READ (N, 12) VAL, COST, PRICE 120 READ (IUNIT) (T (J), J=1, N), S
<b>REAL</b>	100 REAL A(10), K, NET(6,2,2)
<b>RETURN</b>	100 RETURN
<b>REWIND</b>	100 REWIND 110 REWIND 2 120 REWIND KRAK
<b>STOP</b>	100 STOP 110 STOP "UGH" 120 STOP V
<b>SUBROUTINE</b>	100 SUBROUTINE KRUG 110 SUBROUTINE LIMP(V,W,X)
<b>\$USE</b>	100 \$USE EXCEPN 110 \$USE MATRIX*
<b>WRITE (temporary file)</b>	100 WRITE 16, B, D 110 WRITE "FIRST TABLE VALUE" 120 WRITE, (T(K), K=1, 25), X, X 3
<b>WRITE (...)</b> (permanent file)	100 WRITE (3) "MONTHLY SUMMARY" 110 WRITE (3, 12) ↑↑LNO, " REPORT TO DATE", 120 + F1, F2, F4, F7

- Uses FORTRAN-IV style I/O statements
- Terminal I/O; temporary and permanent file I/O
- May use unformatted input and standard formal output (including list-directed slewing); standard format output is for terminal only
- Or may use extensive formatting facilities:

Alphabetic	Octal
Exponential	Power
Fixed-point	Type (substituted type)
General (E or F)	X (blanks)
Hollerith	* (substituted number)
Integer	" (quoted rather than counted Hollerith)



## FORTRAN LANGUAGE (Continued)

### SUBPROGRAMS

- **Intrinsic** Accepts all FORTRAN II closed and open functions with either FORTRAN II or FORTRAN IV names
- **External** FUNCTION, SUBROUTINE, main program
- **Internal** Arithmetic Statement Function  
Internal Functions (a generalization of ASF)
- **Entry** ENTRY statement allows multiple entry to external and internal subprograms and to main program

### MODES (INTEGER OR REAL)

- **Implied** If first letter of user-supplied name is I, J, K, L, M, or N, the named variable, array, or subprogram has implied mode INTEGER; otherwise it has implied mode REAL.
- **Assumed** If an optional single mode \$OPT REAL or \$OPT INTEGER is given all user-named items are assumed to have that mode.
- **Declared** Items named in INTEGER or REAL statements (or INTEGER or REAL qualified declarations or subprogram definition statements) have declared mode whether or not it agrees with implied or assumed mode.
- **Mixed** Integer values used in real calculations are automatically converted to real form; and, real values used in a context where an integer value is required are automatically truncated.

### OPERATORS

Operator Symbol	Operation Specified
=	Assignment
+	Addition
-	Subtraction or negation
*	Multiplication
/	Division
** or ↑	Exponentiation

### CONSTANTS

Integers within the range:  $-524287 \leq \text{integer} \leq 524287$

Reals within the range:  $.863616852 \times 10^{-77} \leq (\text{real}) \leq .57896044 \times 10^{77}$

Octal integers within the range:  $0 \leq \text{octal integer} \leq 3777777_8$

Quoted characters from one to three characters

### NAMES

Composed from letters, digits and \$.  
First character must be a letter.  
May be from 1-30 characters in length.  
Blanks are ignored.

### ARRAYS

May have from 1-15 dimension arrays

**GENERAL  ELECTRIC**

7735 OLD GEORGETOWN ROAD, BETHESDA, MARYLAND 20014

118058 (75M) 11/68

NORM THURSTON 663-3860