

x 6818.2013

micromachines

Ine

THE COMPUTER HISTORY MUSEUM



1 027 7670 2

F. GERALD SNYDER



**TELEDYNE  
SYSTEMS COMPANY**

19601 NORDHOFF STREET

NORTHRIDGE, CALIFORNIA 91324

(213) 886-2211 TWX (910) 493-1249

MERS KUTT

KUTT SYSTEMS, INC.

11 RUDEN CR.

DOUG BILLS, OWNER

(416) 449-~~5853~~

# MCM

MICRO COMPUTER  
MACHINES INC  
4 LANSING SQUARE  
WILLOWDALE  
ONTARIO CANADA  
M2J 1T1  
TEL 416/492 1693

# MCM/70

## DESK TOP COMPUTER



The MCM/70 is a modular family of versatile and powerful stand-alone computers. They are of a size, price and ease-of-use as to bring personal computer ownership to business, education and scientific users previously unserved by the computer industry.

The MCM/70 breaks through the barriers that have previously inhibited widespread computer usage. Traditionally, computers have been expensive and difficult to operate directly without extensive training. MCM/70 APL computers are inexpensive and easy to use. Virtually any executive or professional, without any prior computer background, can begin using the MCM/70 effectively within an hour.

MCM/70 APL computers have extremely broad applications in business, education and science for such activities as:-

accounting and data processing  
word processing  
problem solving

communications  
computer assisted learning  
data entry and collection

The standard unit incorporates a central processing unit with full MCM/APL, full typewriter and APL keyboard and a plasma TV-like 32 character display screen. Its performance may be greatly enhanced with the addition of direct-access cassettes; diskettes; impact printers; data communication adapters and CRT displays. A freely-programmable stand-alone computer, including keyboard and display, will sell for approximately \$3500.

### BASIC CONFIGURATION SPECIFICATIONS

- |  |   |
|--|---|
| <p>a) <b>INPUT</b><br/>APL Keyboard<br/>- 46 keys (IBM 2741 layout) plus start key and space bar<br/>- 2 control keys (control, shift)</p>   | <p>d) <b>MCM/APL</b><br/>APL/360 Compatible<br/>Extensions Include<br/>- execute primitive<br/>- dyadic null primitive<br/>- advanced function editing</p>                      |
| <p>b) <b>OUTPUT</b><br/>1 Line Plasma Display<br/>- 32 characters per line<br/>- 5x7 dot matrix characters<br/>- 80 character I/O buffer</p>   | <p>Precision<br/>- 16 significant digits internally<br/>- up to 16 displayed (keyboard control)<br/>Range: <math>-8 \times 10^{75}</math> to <math>+7 \times 10^{75}</math></p> |
| <p>c) <b>INTERNAL COMPUTER</b><br/>Microprogrammed APL Machine<br/>- 2k user work area (expandable to 8k)<br/>- 120k virtual memory cassette system option<br/>Data Storage Types<br/>- characters - 1 byte ea.<br/>- integer numbers - 1 to 8 bytes<br/>- decimal numbers - 8 bytes ea.</p> | <p>e) <b>PHYSICAL CHARACTERISTICS</b><br/>Basic Chassis Size: 13" W<br/>15-3/4" D<br/>5-1/4" H<br/>Weight: 20 lbs.<br/>Power: 120V 60 hz or 220V 50hz<br/>50 watts</p>          |

OPTIONS AVAILABLE

a) *MAGNETIC TAPE CASSETTE*

1 or 2 cassette drives may be integrated into the standard chassis.

Record size - 128 bytes  
Storage capacity - 120,000 bytes  
Tape search speed - 40 IPS  
Read/write speed - 10 IPS  
Transfer rate - 700 CPS  
Read after write  
Byte parity  
Record check sum  
Weight: 2 lbs.

b) *IMPACT PRINTER*

Plug-in module in a separate chassis  
Chassis includes a numeric keypad  
Characters: 5 x 7 dot matrix  
APL, alphabetic capitals, numerics  
Size: .075" W x .105" H  
10 deg. slant from the vertical  
Line width: 34 characters  
Line spacing: 6 lines/in.  
Print speed: 110 cps  
Max. line speed: 4 lines/sec  
Ribbon: red/black ink  
Paper: standard adding machine roll  
Weight: 7.5 lbs.  
Power: 45 watts

c) *OTHER OPTIONS*

- 1 - EIA compatible communications interface - keyboard selection of data rates up to 9600 baud (asynchronous)
- 2 - CRT display - 16 lines of 64 characters ea.
- 3 - 132 column hard copy printer
- 4 - Battery power operation
- 5 - Diskette drives

PRIMITIVE FUNCTIONS

SCALARS

Monadic

+Y	Y
-Y	0-Y
*Y	Signum Y
÷Y	Reciprocal of Y
*Y	e to the Yth power
⌈Y	Ceiling of Y
⌊Y	Floor of Y
Y	Absolute value of Y
⊙Y	Natural logarithm of Y
!Y	Factorial Y
∘Y	Pi times Y
?Y	A random number from 1 to Y
~Y	Not Y

Dyadic

X+Y	X plus Y
X-Y	X minus Y
X×Y	X times Y
X÷Y	X divided by Y
X*Y	X to the Yth power
X⌈Y	Maximum of X and Y
X⌊Y	Minimum of X and Y
X Y	X-residue of Y
X⊙Y	Base-X logarithm of Y
X!Y	Binomial coefficient: Y items taken X at a time.
XOY	See trigonometric functions
X<Y	X less than Y
X≤Y	X less than or equal to Y
X=Y	X equal to Y
X≥Y	X greater than or equal to Y
X>Y	X greater than Y
X≠Y	X not equal to Y
X∨Y	X or Y
X∧Y	X and Y
X∨Y	Neither X nor Y
X∗Y	Not both X and Y (X nand Y)

TRIGONOMETRIC FUNCTIONS

Y is in radians			
10Y	Sine Y	~10Y	Arcsin Y
20Y	Cosine Y	~20Y	Arccos Y
30Y	Tangent Y	~30Y	Arctan Y
40Y	(1+Y*2)*.5	~40Y	(~1+Y*2)*.5
50Y	Sinh Y	~50Y	Arcsinh Y
60Y	Cosh Y	~60Y	Arccosh Y
70Y	Tanh Y	~70Y	Arctanh Y
00Y	(1-Y*2)*.5		

MIXED

ρY	Dimension of Y
XρY	Reshape Y to have dimension(s) X
ιY	First Y consecutive integers from origin (0 or 1)
XιY	First location(s) of Y within vector X
X[Y]	Yth element(s) of X
XεY	Membership of X in Y
XτY	Representation of Y in number system X
X⊥Y	Value of the representation Y in number system X
X?Y	X integers selected randomly without replacement from 1 to Y
ϕ[N]Y	Reversal along the Nth dimension of Y
⊖Y	Reversal along the first dimension of Y
Xϕ[N]Y	Rotation by X along the Nth dimension of Y
X⊖Y	Rotation by X along the first dimension of Y
⊖Y	Transpose Y
X⊖Y	Transpose Y according to X
.Y	Ravel Y (make Y a vector)
X,Y	Catenate Y to the last dimension of X
X,[N]Y	Catenate Y to the Nth dimension of X
X+Y	Take first or last  X  elements of Y as X is + or -
X-Y	Drop first or last  X  elements of Y as X is + or -
ΔY	Indices of values of the vector Y sorted in ascending sequence
∇Y	Indices of values of the vector Y sorted in descending sequence
⚡Y	Execute the literal vector Y as an APL statement

GENERALIZED SCALAR DYADIC

In the entries below, the symbol ⊙ (not an APL symbol) represents any scalar dyadic function.

X⊙.⊙Y	Generalized inner product of X and Y
X⊙.⊖Y	Generalized outer product of X and Y
⊖/Y	Generalized reduction along the last dimension of Y
⊖/[N]Y	Generalized reduction along the Nth dimension of Y
⊖/Y	Generalized reduction along the first dimension of Y
⊖\Y	Generalized scan along the last dimension of Y
⊖\[N]Y	Generalized scan along the Nth dimension of Y
⊖\Y	Generalized scan along the first dimension of Y

COMPRESSION AND EXPANSION

X/Y	(logical) compression along the last dimension of Y
X/[N]Y	(logical) compression along the Nth dimension of Y
X/Y	(logical) compression along the first dimension of Y
X\Y	(logical) expansion along the last dimension of Y
X\[N]Y	(logical) expansion along the Nth dimension of Y
X\Y	(logical) expansion along the first dimension of Y

## SYMBOLS

( )	Parentheses for nesting
+	Branch
←	Assign data to a variable
□	Quad for input and output
▣	Quote-quad for literal input
'	Quote for literal data
⋈	Comment indicator
▽	Del for mode change
⊕	Lock a defined function
⊖	(O-U-T) Halt literal input request
-	Negative sign
E	Exponential notation
.	Period or decimal point
:	Label separator
;	Output and index separator
\$	Dollar sign
°	Statement separator
_	Underline

## SYSTEM FUNCTIONS

□EX A	Expunge or erase objects named by rows of A, except groups, labels, or halted functions. (A may be a vector if only one object is to be erased)
□FN	Returns matrix result in which each row contains the name of a function currently in the workspace.
□OF	Terminate current session.
□VA	Returns matrix whose rows contain names of all global variables along with any local variables and label names of halted functions.
□WC	Clears entire contents of workspace.

## SYSTEM VARIABLES

□CT	Comparison tolerance: for the functions [ < ≤ ≥ > ≠
□IO	Index origin: 0 or 1
□LC	Line counter: line numbers of functions in execution innermost first
□PP	Print precision: 2 to 16 significant digits
□PT	Print time: display latency measured in quarters of a second
□PW	Printing width: 32 to 128 characters on the MCM/70 dependent on the output device
□RL	Random link: seed for generating random numbers
□SI	State indicator: N by 3 matrix with the rows containing the names of functions in execution innermost first
□SO	Diverted source: accept input from a peripheral device (1 to 199)
□SN	Diverted sink: direct output to a peripheral device (1 to 199)
□WA	Workspace available

## ERROR MESSAGES

Message	Meaning
DOMAIN ERROR	Function not defined for given values of the arguments
INDEX ERROR	Indexing a non-existing dimension of an array
LENGTH ERROR	Operation asked to be performed on arrays with nonconformable coordinate lengths
RANGE ERROR	The result of the function performed cannot be represented
RANK ERROR	Nonconforming or inappropriate array structure
SYNTAX ERROR	Missing argument Unbalanced parentheses Two juxtaposed variables with no operator Uneven number of quotation marks
TAPE ERROR	Cannot physically access the tape cassette
VALUE ERROR	Reference to an undefined variable
WS·FULL	Instructions required more workspace than is available

## MCM/APL KEYBOARD



# PROGRAMMING LANGUAGES COMPARISON

computing the average of a set of numbers

BASIC

```
10 DIM X (100)
20 LET S=0
30 READ N
40 FOR I=1 to N
50 READ X(I)
60 LET S=S+X(I)
70 NEXT I
80 LET A=S/N
90 PRINT A
100 DATA
    - - - - -
    - - - - -
XXX END
```

FORTRAN

```
      DIMENSION X (100)
      READ (5,10) N, (X(I),I=1,N)
10    FORMAT (I5/(F10.0))
      S=0.0
      DO 15 I=1,N
15    S=S+X(I)
      A=S/N
      WRITE (6,20)A
20    FORMAT (E15.2)
      END
```

APL

▽ AVG X  
[1] (+/X)÷ρX←□  
[2] ▽

sorting a set of numbers

BASIC

```
10 DIM X (100), Y(100)
20 READ N
30 FOR I=1 to N
40 READ X (I)
50 NEXT I
60 FOR I=1 to N
70 LET A=X(I)
80 LET L=1
90 FOR J=1 to N
100 IF A <=X(J) THEN 130
110 LET A=X(J)
120 LET L=J
130 NEXT J
140 LET Y(I)=A
150 LET X(L)=100000.
160 PRINT Y(I)
170 NEXT I
180 DATA
    - - - - -
    - - - - -
    - - - - -
XXX END
```

FORTRAN

```
      DIMENSION X (100), Y(100)
      READ (5,10) N, (X(I),I=1,N)
10    FORMAT (I5/(F10.2))
      DO 20 I=1,N
      A=X(I)
      L=1
      DO 15 J=1,N
      IF (A-X(J)) 15,15,12
12    A=X(J)
      L=J
15    CONTINUE
      Y (I)=A
20    X (L)=100000.
      WRITE (6,30) (Y(I),I=1,N)
30    FORMAT (E15.2)
      END
```

APL

X[ΔX←□]



## THE MCM/70 IN SCIENCE

The *MCM/APL* computer offers both the scientist and the engineer a concise, straightforward method for solving problems. These problems may vary from electrical analysis to evaluating stress data for bridge construction. Below is a sample programme called *DSTAT* which performs various statistical computations. Notice that each line performs, with ease, a different calculation.

A Statistical Programme Called *DSTAT*:

```

▽ DSTAT X;MEAN;VAR;V;W;N
[1] 'SAMPLE SIZE           ';N←ρX←X[ΔX]
[2] 'MAXIMUM               ';X[N]
[3] 'MINIMUM               ';X[1]
[4] 'RANGE                 ';X[N]-X[1]
[5] 'MEAN                  ';MEAN←(+/X)÷N
[6] 'VARIANCE              ';VAR←(+/(X-MEAN)*2)÷N-1
[7] 'STANDARD DEVIATION   ';VAR*0.5
[8] 'MEAN DEVIATION       ';(+/|X-MEAN)÷N
[9] 'MEDIAN                 ';0.5×X[⌈N÷2⌋]+X[1+⌊N÷2⌋]
▽

```

An Example Of Its Use With Sample Data Of 3 2 4.5 7 6 3 1 5.75 1.1

```

DSTAT 3 2 4.5 7 6 5 3 1 5.75 1.1

SAMPLE SIZE           10
MAXIMUM               7
MINIMUM               1
RANGE                 6
MEAN                  3.835
VARIANCE              4.494472222
STANDARD DEVIATION   2.120017033
MEAN DEVIATION       1.815
MEDIAN                 3.75

```



## THE MCM/70 IN BUSINESS

```
INVENTORY← 11 22 33 44 55
RECEIPTS← 23 34 45 56 67
SALES← 25 35 50 65 75
PRICE← 9.99 4.89 6.59 2.49 5.99
NEW←INVENTORY+RECEIPTS-SALES
REVENUE←SALES×PRICE
```

The above example illustrates how easily *MCM/APL* may be applied to a typical business inventory problem. The symbols used to perform the calculations are familiar mathematical symbols  $+ - \times$ , with the exception of the left pointing arrow  $\leftarrow$ . By using this one additional symbol, all the necessary data can be entered and stored in the computer's memory under names arbitrarily chosen by the user. These names can then be used in any of the ensuing computations.

The example takes the current inventory of five stock items and stores them in memory under the name *INVENTORY*. It also stores receipts, sales and unit prices under the names *RECEIPTS*, *SALES* and *PRICE*. Line 5 updates the *INVENTORY* to reflect *RECEIPTS* and *SALES*. The new inventory levels are stored in memory under the name *NEW*. These new inventory levels can be displayed by typing the name of their storage location - *NEW*.

```
NEW
9 21 28 35 47
```

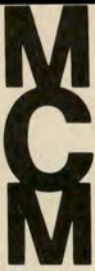
Line 6 computes the revenue derived from the current sales and stores them under the name *REVENUE*. The values stored in *REVENUE*, like those in *NEW*, are displayed by typing that name.

```
REVENUE
249.75 171.15 329.5 161.85 449.25
```

The total revenue received from sales can be obtained by using the slash symbol ( $/$ ) in conjunction with the plus sign in the following manner:

```
+/REVENUE
1361.5
```

With this same data, a wide variety of other computations such as sales to inventory ratios, percentage of sales by product, and others can be readily performed without learning any more new symbols. By learning the functions of the complete set of more than 50 such symbols available in the *MCM/70*, the applications possible are virtually limitless.



# THE MCM/70 IN EDUCATION

The MCM/70 Desk Top computer represents a breakthrough in technology of tremendous significance to education.

The major difficulty in advancing the use of computers in education has been that the technology as it exists today is simply too expensive. The most effective uses of computers in educational environments have involved some form of advanced terminal based system -- systems that put the student into direct and interactive communication with the computer -- but the size of computer required and the communications facilities that terminal systems demand makes it virtually impossible to provide adequate facilities for the large masses of students enrolled in our school systems. This means that, in general, educators have had to accept second best or inadequate facilities, in many instances limiting introduction on any broad scale.

The MCM/70 breaks through this barrier and makes it possible, for less than the cost of what might otherwise only pay for the terminals, to provide each student in a computer equipped classroom with his own individualized interactive computer.

The interactive mode in which the MCM/70 operates is significantly supported by the programming language it employs, the programming language called APL. APL, which stands for A Programming Language, was first introduced in the mid 1960's as a system of notation for conveying ideas in the classroom rather than for specifically communicating with computers. Since that time its use (in business, and education) has spread rapidly worldwide. At one and the same time, APL is the simplest and most powerful programming language yet developed. While it is being taught in some schools to 10 and 11 year olds, at the same time it is the language used most by IBM for internal interactive programming work. It is a very simple language to learn. It is a very concise language, avoiding many of the grammatical complexities of other programming languages. It is a very powerful language, capable of handling problems that wouldn't be attempted with other languages. These features make APL the most effective conversational language in the computer world today, and the MCM/70, with its full implementation of APL, a most significant advance in the development of interactive computer technology.

Featuring a language suitable in its simplicity and power for students and teachers alike, the MCM/70 brings to the world of education a technological solution to the problem of introducing economical interactive computer systems.

<u>INTERACTIVE COMPUTING IN THE CLASSROOM</u>	<u>Time Sharing</u>	
<u>THE ECONOMIES OF THE MCM/70</u>	<u>Systems</u>	<u>MCM/70</u>
Cost of Basic Terminal	Yes	Yes
Cost of Installation, Wiring, etc.	Yes	No
Cost of Data Sets	Yes	No
Cost of Line Charges	Yes	No
Cost of Connect Time	Yes	No
Cost of CPU Time	Yes	No
Cost of Data Storage	Yes	No
AVERAGE COST PER STUDENT HOUR	About 25 Dollars	About 25 Cents



## MICRO COMPUTER MACHINES INC.

### BACKGROUND INFORMATION

*Micro Computer Machines Inc. was formed in late 1971 for the purpose of developing the MCM/70, a revolutionary concept in the microminiaturization of computers.*

*MCM is a Canadian company as are all of its employees. The founder and president, Mers Kutt, is one of the best known personalities on the Canadian computer scene, having spent over 15 years in the business and educational computer fields. A former Professor of Computer Science at Queen's University, and president of the Canadian Information Processing Society, he is probably best known for being the entrepreneur-developer and first president of Consolidated Computer Inc., Canada's only computer company with a world-wide reputation.*

*The company head office is in Toronto, Ontario and the development lab is in Kingston, Ontario.*

### PRODUCT LINE

*The revolutionary concept of the MCM/70 is that it brings to the world of computing what the \$100 hand held calculator brought to the world of calculating. In a desk top model smaller than an office typewriter, the MCM/70 offers a portable, self-contained, general purpose computer. Weighing less than 20 lbs. and capable of being operated in any office environment by simply plugging it into a standard electrical outlet, the MCM/70 offers the user a micro computer with a full implementation of the APL programming language to handle a diverse set of business and scientific problems. APL, a very simple yet extremely powerful computer language, is rapidly becoming the most sought after language in computing - both by non-computer oriented people, as well as seasoned programmers. Priced at under \$3500 U.S. for the basic unit, the MCM/70 represents a dramatic price breakthrough that opens up an enormous new market for computers. As a self standing unit, applications exist in education, engineering, insurance, as well as research, finance, accounting and as an executive desk top computer. Then by adding various modularly designed peripherals, the unit has wide application as an intelligent terminal or as a small-business data processing system. Unlike the specialized and limited markets to which most computers are restricted, these are markets of considerable and generally untapped depth. We expect that the reputation the MCM/70 builds will be world wide.*

**PRODUCTS OF TOMORROW...READY FOR PROBLEMS OF TODAY**

102776702