UNCI	ASSIFI	5 NAV A P JUN	AL POST NETHOD F	FOR EVAL MASTRA	TE SCHOOLUATION	OL MON OF MIC	TEREY C Rocompu	A TERS FO	R TACTI	CAL APP	F/G 9 LICAT-	9/2 ETC(V)	
	د : ***	іл. Ф.л. с											
										. <u>.</u> . (0	ţ		





2514 0

REPO	RT DOCUMENTA	TION PAC	E	READ INSTRUCTIONS
REPORT NUMBER		2. 0		1. RECIPIENT'S CATALOG NUMBER
		11	D-A09109:	5
TITLE (and Sublitio)				S. TYPE OF REPORT & PERIOD COVER
A Method for	Evaluation	of Micr	ocomputers	Master's Thesis:
for Tactical	Application	S		June 1980
				6. PERFORMING ORG. REPORT NUMBER
AUTHORIE				S. CONTRACT OR GRANT NUMBER(A)
Marcos Mastra	kaa			
Maicus Mastia	NG3			
PERFORMING ORGANI	TTION NAME AND A	DORESS		18. PROGRAM ELEMENT, PROJECT, TAS
Naval Postgra	duate Schoo	1	/	AREA & WORK UNIT NUMBERS
Monterey, Cal	ifornia 939	40		
Navol Doctors	E NAME AND ADDRE	(58) . 1		TUDA 1080
Monterey Cal	ifornia 070	40		
monterey, tal	TTALINIA 233	τU		189
MONITORING AGENCY		I different free	Controlling Office)	18. SECURITY CLASS. (of this report)
Naval Postgra	duate Schoo	1		Unclassified
Monterey, Cal	ifornia 939	40	-	ISE. BECLASSIFICATION/DOWNGRADING
				SCHEDULE
Approved for	public rele	ase; di	stribution	unlimited
Approved for	public rele	ase; di	stribution	unlimited
Approved for Distribution state	public rele	ase; di	stribution	unlimited
Approved for DISTRIBUTION STATE SUPPLEMENTARY NO MICTOPTOCESS	public rele MENT (a) Ika american TES	ease; di	stribution	unlimited
Approved for Distribution state Supplementary no Microprocess Microprocess	public rele MENT (a) the energed TES	ease; di onlored in Bi Tactica syste	stribution	unlimited <i>Report</i> ) <i>Report</i> ) ion of microcomputer
Approved for DISTRIBUTION STATE SUPPLEMENTARY NO Microprocess Microcompute	public rele MENT (a) the energies TES TES	ease; di contored in Bi contored in Bi cont	stribution	unlimited
Approved for Distribution state Supplementary no Microprocess Microcompute Multi-microc	public rele MENT (af IRe enerrow TES TES cor er computer	ease; di entered in Bi Tactica Syste Perform	stribution sot 30, 11 different for nilly by block number l applicat: ms ance capac:	unlimited <i>Report</i> ) ion of microcomputer ity of computers
Approved for Distribution state Supplementary no Microprocess Microcompute Multi-microc	public rele MENT (of the energed TES TES computer	ease; di contored in Br Tactica syste Perform	stribution •••• 30, // different & •••• 30, // different & ••••• 30, // different & •••••• 30, // different & ••••• 30, // different & ••••• 30, // different & •••••• 30, // different & •••••• 30, // different & •••••• 30, // different & ••••••••••••••••••••••••••••••••••••	unlimited <i>Report</i> ) ion of microcomputer ity of computers
Approved for DISTRIBUTION STATE SUPPLEMENTARY NO Microprocess Microcompute Multi-microc ABSTRACT (Continue A method computer syst	public rele MENT (a) the energies TES TES TES TES TES TES TES TES TES TES	ease; di ease; di ease; di ease; di ease; di factica syste Perform to eva ecified	stribution set 30, 11 different & notify by block number l applicat: ms sance capac: with by block number luate the p tactical a	unlimited <i>Report</i> ) ion of microcomputer ity of computers performance of micro- upplication. The
Approved for DISTRIBUTION STATE SUPPLEMENTARY NO Microprocess Microcompute Multi-microc Amethod computer syst computational in terms of p microcomputer performance c	public rele	ease; di contored in Bi contored in Bi Tactica Syste Perform to eva ecified its of a paramet microco re also	stribution	" " " " " " " " " " " " " "
Approved for DISTRIBUTION STATE SUPPLEMENTARY NO SUPPLEMENTARY NO Microprocess Microcompute Multi-microc Amethod computer syst computational in terms of p microcomputer performance c parameters.	public rele MENT (a) the environ TES TES TES TES TES TES TES TES TES TES	ease; di control of a control of a control of a control of a paramet microco re also cormance	stribution	unlimited <i>Report</i> ion of microcomputer ity of computers performance of micro- upplication. The upplication are specific presently marketed tems computational in terms of performance s not less than the
Approved for DISTRIBUTION STATE SUPPLEMENTARY NO Microprocess Microcompute Multi-microc AUSTRACT (Continue of Multi-microc AUSTRACT (Continue of AUSTRACT (Continue of AUSTRACT (Continue of AUSTRACT (Continue of Multi-microc AUSTRACT (Continue of Multi-microc AUSTRACT (Continue of Multi-microc AUSTRACT (Continue of AUSTRACT (Continue of Multi-microc AUSTRACT (Continue of AUSTRACT (Continue of Multi-microc AUSTRACT (Continue of AUSTRACT (Continue of Multi-microc AUSTRACT (Continue of AUSTRACT (Cont	public rele MENT (a) the energy TES TES TES TES TES TES TES TES TES TES	ease; di controred in Bi controred in Bi Tactica Syste Perform to eva ecified its of a paramet microco ire also iormance	stribution 	unlimited """"""""""""""""""""""""""""""""""""
Approved for DISTRIBUTION STATE SUPPLEMENTARY NO SUPPLEMENTARY NO Microprocess Microcompute Multi-microc Amethod computer syst computer syst computational in terms of p microcomputer performance co parameters. "ORM" 1473 co age 1)	public rele MENT (a) the energy TES TES TES TES TES TES TES TES TES TES	ease; di ase;	stribution stribution set 30. // different & intry by block number l applicat: ms ance capac: why by block number luate the p tactical a ers. The p mputer syst specified capacity i UN secumity C	unlimited <i>Report</i> ion of microcomputer ity of computers performance of micro- upplication. The upplication are specific presently marketed tems computational in terms of performance s not less than the NCLASSIFIED AND FICATION OF THIS PACE (Ther Date of

performance requireme evaluated as a feasil tactical application A6-E tactical system	ent, then the microcomputer system is ble system suitable for implementing the . A case study using the attack aircraf illustrates the evaluation method.
•	
Accession For NTIS CRA&I	
DDC TAB	
By	4
Distribution/	
Availand/or Dist. special	
A	
	· · ·
) Form 1472	UNCLASSIFIED

Market Street

4-6-

### Approved for public release; distribution unlimited

A Method for Evaluation of Microcomputers for Tactical Applications

by

Marcos Mastrakas Lieutenant Commander, Hellenic Navy B.S., Hellenic Naval Academy, 1962

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL June 1980

Author

Approved by: Uno R. Fodus Thesis Advisor German Case Second Reader Chairman Department of Computer Sciences Dean ON Phformation and Policy Sciences

## ABSTRACT

A method is proposed to evaluate the performance of microcomputer systems in a specified tactical application. The computational requirements of a tactical application are specified in terms of performance parameters. The presently marketed microcomputer and multi-microcomputer systems computational performance capacities are also specified in terms of performance parameters. If the performance capacity is not less than the performance requirement, then the microcomputer system is evaluated as a feasible system suitable for implementing the tactical application. A case study using the attack aircraft A6-E tactical system illustrates the evaluation method.

Sugar Cherry

# TABLE OF CONTENTS

Ē

I.	INTE	RODUCTION	12
	Α.	GENERAL	12
	Β.	MAJOR ISSUES	13
	с.	STANDARDIZATION	14
	D.	APPLICATION TRENDS AND OPPORTUNITIES	15
	Ε.	COMPOSITION OF THESIS	16
II.	BACI	KGROUND	18
III.	THE Comi	USE AND PERFORMANCE REQUIREMENTS FOR PUTER SYSTEMS IN TACTICAL APPLICATIONS	32
	A.	GENERAL VIEW	32
	В.	SENSOR CONTROL	33
	c.	COMMAND AND DISPLAY	33
	D.	WEAPONS CONTROL	37
	E.	GUN CONTROL	38
	F.	GUIDED MISSILE CONTROL	39
	G.	A CRITICAL QUESTION ABOUT MICROCOMPUTERS	40
IV.	METI	HOD OF EVALUATION	42
	Α.	COMPUTING CAPACITY OF UNIPROCESSORS	42
	Β.	COMPUTING CAPACITY OF MULTI-MICROCOMPUTER SYSTEMS	45
	с.	ADDITIONAL IMPORTANT CRITERIA	53
	D.	DESIRABLE FEATURES	69
v.	CASI	E STUDY	91
	A.	THE A6-E OPERATIONAL FLIGHT PROGRAM	91
	B.	PERFORMANCE ESTIMATES FOR UNIPROCESSOR	108

5

Carlos and the second

C.	. PER Sys	FORMANCE ESTIMATES FOR MULTI-COMPUTER	114
D.	. SUM	MARY	117
VI. CO	ONCLUS	IONS	118
APPENDIX	X A:	Required Time to Select a Microcomputer	122
APPENDIX	Х В:	Checklist for Evaluating Microcomputer Vendor Policies	127
APPENDIX	Х С:	An Interactive Computer Graphics (ICG) System Selection Checklist	134
APPENDIX	X D:	Useful Tables	135
GLOSSARY	Y		160
LIST OF	REFER	RENCES	186
INITIAL	DISTR	IBUTION LIST	189

A second

الانشار

# LIST OF FIGURES

1.	Block diagram of a microprocessor/microcomputer	19
2.	A single board Z8000 microcomputer	20
3.	Magnetic bubble memory	22
4.	Characteristics of different kind of memories	24
5.	Comparison between MBMs and other memories	25
6.	Advantages and disadvantages of multi- microcomputer systems	29
7.	Comparison of execution times	31
8.	AEGIS ship combat system, new technology applications	34
9.	AEGIS ship combat system baseline	35
10.	A multi-microcomputer system	51
11.	A genealogy of current languages for microprocessors	57
12.	Relationship of O.S. to basic computer hardware	66
13.	Real-time operating system conceptual approach	67
14.	Overview of Kernel layering for general real-time operating system	67
15.	Life-cycle cost	83
16.	Examples of potential application areas	89
17.	Comparison of two potential distributed computer systems	89
18.	A typical multi-microcomputer system]	14
19.	Layers of the operating system1	36

7

A State of the second s

# LIST OF TABLES

1.	Different Types of Computers	30
II.	Performance Capacities of Different Computer Systems	44
IIa.	Three Commonly Used Mixes	44
IIb.	Instruction Times (in µs) for Different Computers	46
IIc.	Performance Capacities of Different Computer Systems	50
III.	Information and Characteristics of Languages MPL, PL/M, PLZ-SYS	60
IIIa.	General Information	60
IIIb.	Minimum Memory Requirements	60
IIIc.	Simple Data Types	60
IIId.	A Spectrum of Language Processing Solutions	61
IIIe.	Control Structures	61
IIIf.	Procedures and Parameter Passing	62
IIIg.	Comparison of Languages Processing Solutions	62
IV.	Details of Operational Flight Program, of the A6-E Aircraft	94
v.	Execution Time Estimates for Each Major Function-	112
VI.	Uni-Microprocessor Performance Evaluation	113
VII.	Functional Partitioning of the A6-E Operational Flight Program	116
VIII.	Required Time to Select a $\mu$ C	123
IX.	General-Purpose Microprocessors	138
<b>x.</b>	All-In-One Processors	140
XI.	Bit-Slice Families	143

i.

XII.	Directory of µPs by Vendor	144
XIII.	Bubble-Memory Devices	146
XIV.	Operating Systems for Microcomputers	147
xv.	Semiconductor Technologies	151
XVI.	μΡ/μC-Chip Families	154
XVII.	Production Volumes of $\mu Ps$ and $\mu Cs$	155
XVIII.	List of Original Source µP/µC Manufacturers	156

"In its thirty years of experience with computers, the Navy has discovered--and often rediscovered--one fundamental truth: that people are the vital key to success in the computer age!"

> Rear Admiral Frank S. Haak, USN "Brainware versus Hardware" [1976]

#### ACKNOWLEDGEMENT

I would like to thank Professor Uno Kodres, who led me in the mysteries of microprocessors/microcomputers. He helped more than his share on this thesis, and is more than an advisor to me.

The second

#### I. INTRODUCTION

A. GENERAL

The microcomputer technology, which came into existence in 1971, has grown from a low performance four binary digit word length device (Intel 4004) to a high performance sixteen binary digit word length computer, the Motorola MC 68000, in 1979.

The central processing unit (CPU) of the MC 68000 is capable of addressing 16 Megabytes of memory, which is twice the memory capacity of the CRAY-1, the world's fastest and largest computer.

A thirty-two bit word length microcomputer system with sophisticated architecture is soon to be announced. Clearly, the advancements in Large Scale Integrated (LSI) and Very Large Scale Integrated (VLSI) technologies are coming so rapidly that just to keep up with the newest information is becoming a challenge.

The Armed Forces, and the U.S. Navy in particular, have long been pioneers in the development of new systems. The development of large computer systems normally lags behind the state of the art because of the length of development time. Microcomputer systems are virtually nonexistent in tactical applications. This may be attributed to a lack of knowledge of their capabilities and to the rapid advancement of microcomputer technology, coupled with the high risk of trying something new.

12

The intent of this thesis is to give the user the necessary information and background in the area of microprocessors and microcomputer systems and also to provide a method to evaluate microcomputer systems for tactical applications, i.e., Command Control Communications ( $C^3$ ), weapon systems, etc. At the moment the only known systematic method for the evaluation of computers for military use is the Military Computer Family (MCF) report in 1976 [3].

#### **B. MAJOR ISSUES**

There are several major issues.

1. Processing Needs

Given an application described by parameters, such as: (1) Number/per sec of loads, stores, compares.

(2) Number/per sec of additions, subtractions, etc.

(3) Number/per sec of floating point operations.

- (4) Number/per sec of multiplies, divides.
- (5) Number of instructions/bytes in the program and data.
  - 2. <u>Processing Capacity of LSI/VLSI Systems</u> Stated in terms of parameters:

a. Uniprocessor

- (1) Number/per sec of stores, loads, compares.
- (2) Number/per sec of additions, subtractions, etc.
- (3) Number/per sec of multiplies, divides.
- (4) Number/per sec of floating point operations.
- (5) Memory sizes: main + auxiliary memory.

b. Multiprocessor

Multi-single board computer on a Multibus [Intel 77].

- 3. Program Development Capability
  - a. Program Development Tools
- Hosted on large systems. The High Order Language (HOL) Ada used as a program development tool for real-time processing including microprocessor applications.
- (2) Hosted on small development systems.
  - 4. Selection Procedure

Selection of one or more processor families which will provide long term support for a large class of Navy applications.

## C. STANDARDIZATION

Standards for microprocessors are becoming increasingly important for military and industrial market expansion. However, the standards must be objectively chosen with applications engineers and user communities involved in the decisions and must be "technology transparent" to provide continuity and support for microcomputer development without stifling innovation.

The idea of a standard military microprocessor architecture has the advantages of volume purchasing power and logistic support cost savings; the disadvantage appears to be a risk of not following the commercial mainstream.

The Defense Department's Ada HOL is being developed to serve programming needs for real-time processing including microprocessor applications, and is an example of one potential microprocessor HOL standard [26].

#### D. APPLICATION TRENDS AND OPPORTUNITIES

The opportunities have never been greater for microprocessors--in computation, control, logic replacement and special functions for industrial processing and defense systems.

However, a variety of barriers discourages use of microprocessors in many of the lower-volume application areas [26].

The three key barriers to rapid introduction of microprocessors into military, in particular, the Navy systems relate to:

 Concerns for life-cycle supportability (for both military and industrial applications where anticipated system life is 15 years or longer).

2. The belief that the microcomputer performance in military applications is not adequate.

3. The microcomputer technology is not compatible with previously defined architectural standards. Because of large prior investments in software tools and applications software geared to the established hardware standards, there is reluctance to undertake costly changes.

In terms of user requirements, DOD and industry appear to have more in common than is generally perceived. Some military systems do require radiation hardened devices. Fortunately, the two primary approaches to radiation hardening-- $I^2L$  and,

to a lesser extent, silicon on sapphire--may also provide performance and economic advantages. However, a number of military applications, such as radar and electronic countermeasures, require very-high-speed signal processing in contrast to the slower, general-purpose computation more commonly found in commercial applications.

E. COMPOSITION OF THESIS

Section II presents a survey on the characteristics of microprocessors/microcomputers, magnetic bubble memories, Large Scale Integrated (LSI) and Very Large Scale Integrated (VLSI) technologies, information about Department of Defense (DOD) project Very High Speed Integrated Circuit (VHSIC), and some characteristics of different types of computers.

Section III presents a discussion of the use of computer systems in tactical applications (sensor control,  $C^3$ , weapons, etc.), and the possibilities of using microcomputers instead of the standard computer systems, i.e., AN/UYK-7,14,20, IBM-370/168, etc.

Section IV describes a method to evaluate the performance of microcomputer systems in a specified tactical application. It also gives to the evaluator the necessary tools in terms of military requirements, software materials, and how to compute the life cycle cost of a given system.

Section V presents the case study of the attack aircraft A6-E tactical system, in order to illustrate the evaluation method.

Section VI makes a summary and presents certain conclusions of the thesis.

Appendices A through D contain important information to aid the evaluator's work.

Tables I through XVIII contain detailed information about  $\mu P/\mu Cs$ , the available OS for  $\mu Cs$ , etc.

Finally, a glossary explains the terms and definitions about  $\mu P/\mu C$  which are used in the text of this thesis and elsewhere.

It is underlined here that this thesis is based solely on an unclassified bibliography.

#### **II. BACKGROUND**

During the past 25 years the "computer revolution," as in the past the "industrial revolution," has dramatically changed our world and it promises to bring about even greater changes in the near future.

In the middle 1960's the advent of "microcomputers" has both accelerated and expanded the impact of the "computer revolution" [4].

The microprocessor  $(\mu P)$  is one of the latest developments in computer technology. This device has all the functions of the Central Processing Unit (CPU) of a computer on a tiny piece of silicon. Such a device can fetch instructions from a memory, decode and execute them, perform operations (arithmetic and logical), accept data from input devices, and send results to output devices. A microprocessor, together with a memory and input/output devices, forms the "microcomputer" (figures 1, 2).

The microcomputer ( $\mu$ C) represents a very remarkable achievement of engineering ingenuity and industrial know-how at their best. The cost of simple  $\mu$ P chip is a few dollars, and that of a complete  $\mu$ C, having a power similar to mini and maxi computers, is a few thousand dollars.

New technologies, different architectures, and faster memories are having an impact on the computer. From a laboratory curiosity in 1978, the "Magnetic Bubble Memory" (MBM)

Input/Output μC Bus Internal bus Microcomputer Метогу Microprocessor Arithmetic Logic Unit Registers Control Unit

A DAY OF A D

Block diagram of a microprocessor/microcomputer. Figure 1.



とう お子を行われていたい

ŧ

1

Figure 2. A single board 28000 microcomputer.

chip (Fig. 3) has evolved rapidly into a viable commercial device. Likewise, the range of available components has grown faster than anyone expected! New techniques will produce devices using the above chips, with capacity from 1 M bits in 1979 to 4 M bits in 1980 and 16 M bits in 1985-87 (i.e., chip TIB 1000) [9].

Magnetic bubble memories are best suited for use in bulk and auxiliary storage systems. Initial applications have been in microcomputer storage. MBM's are becoming cost competitive with small floppy disks. MBM's have high packing density and can be mounted on the same PC board as the CPU.

Aerospace and military applications will use a significant number of MBM devices, which replace mechanical devices (primarily airborne head-per-track disk units). Due to reliability (i.e., with a  $10^{-10}$  probability of error, mean time between errors is 55 hours, but when corrected can be over 500,000 years!), and nonvolatility (i.e., information is maintained even with power loss), bubble memories are ideal for replacing satellite tape recorders (10 Khr MTBF's) [14]. Due to ruggedness, bubble memories are suitable for tanks, backpacks and other severe environmental/mechanical-stress conditions. MBM's can replace special ruggedized cassettes now used for data acquisition and program loading.

The following figures 4 and 5 give characteristics, advantages and disadvantages of MBM's which presently compare to other memories on the memory hierarchy [14].



Fig 3a Space saver. Availability of INTEL MBM and supporting LSI family of control and driver circuits reduce space, component count for 1Mbit bubble system by order of magnitude, replacing two printed-circuit boards, four 256 K modules, and about 85 ICs.



Fig 3b Complete One Megabit Memory System (shown actual size of 16 square inches) consists of 7110 bubble memory device and support electronics.



23

W. Saltanda Sar

Memory Cherecteristics	Megnetic Bubble Memory	Electron Beam Addressable Memory	Cherge Coupled Device Memory
Storage Representation	Magnetic domain	Electrical charge	Electrical charge
Bit Density	2 × 10 <sup>4</sup> Bits/in. <sup>2</sup>	10 <sup>7</sup> Bits/in. <sup>2</sup>	10 <sup>6</sup> Bits/in. <sup>2</sup>
Access Mode	Sequential	Quasi-random	Sequential or block addressable
Access Time	0.5–10 msec	10-20 <b>µsec</b>	Sequential : 5–50 µsec Block access : 5 µsec
Transfer Rate	100–500 Kbps	4–8 Mbps per tube	1–10 Mbps.
Power Per Bit (During Memory Operation)	0.4 µW	10 <i>µ</i> W	50 µW
Cost Per Bit (1978)	0.2¢	0.005¢	0.05¢
Largest Chip Size Fabricated (1978)	250 Kbit (Texas Instruments, Rockwell International)	120 Mbits per tube 32 (CDC)	64 Kbit (Mnemonics)
Operational Temperature Range	050°C	0-50°C	-10° to 80°C
Succeptibility to Electromagnetic Emanations	Can be shielded	Can be shielded	Can be shielded
Susceptibility to Power Surges	Power must be filtered	Power must be filtered	Power must be filtered

ومواجع ومستعلق ومراجع المعارجة والقارم والمنارية والمتحالة والمتحال والمحاجز والمناجع والمحاج والمحاج والمحاج

Ι.

1

Fig. 4 Characteristics of different kind of memories. (From: "Distributed Micro/ Minicomputer Systems," Cay Weitzman, 1980).

## ADVANTAGES DISADVANTAGES MBMs vs. Floppy Disk 1) Higher reliability 1) Storage media not readily changed 2) Non-mechanical 3) Smaller physical volume 4) Faster access 5) Simpler interface 6) Media integrity MBMs vs. RAM 1) Non-volatile 1) Slower access 2) More bits/device 2) Slower transfer rate 3) Reduced board space MBMs vs. ROM or PROM 1) Programmability 1) Slower access 2) More bits/device 2) Slower transfer rate 3) Reduced board space

Figure 5. Comparison between MBMs and other memories.

Technology has produced, and is continuing to produce, LSI and VLSI (large and very large scale integrated) components of increasing complexity and power. Yet it is difficult to identify the best or even the many ways in which this powerful new tool, VLSI, might be exploited [25].

The design of electronic equipment will change rapidly once VLSI, in the coming years, becomes a key component in the industry [24].

The Pentagon's Very-High-Speed Integrated Circuits (VHSIC) project was initiated by DOD to develop VLSI signal processors with several hundred times higher speed and computing power than today's LSI devices [29].

The goal of the project is pilot production in 1986 of processors containing 250,000 gates, operating at clock speeds of at least 25 MHZ, and performing several million to several billion operations per second. VLSI circuits are also needed to reduce the power consumption, weight, and size of military electronics. These reductions will, in turn, lead to lower life-cycle costs; lower needs for primary power, deck space, air-conditioning, and the like. The costs of primary power and weight in a satellite, for example, are at least \$2000/W and \$5000/Kg, respectively. About \$20 million would be saved on power costs alone by a reduction of lmW in the average operating power per circuit. The cost of integrated circuits, by contrast, is several hundred thousand dollars [29].

26

S. Burney

Military equipment that will use VHSIC are the following:

• SONAR - Acoustic signature analyzers used in the BQQ-5 and the BQQ-6 processing subsystem in strategic (Trident) and attack submarines; the MK-48 processor in homing torpedoes; and in the Proteus processor used in antisubmarine warfare aircraft.

• RADAR - Signal processors for radar systems in the E-2C and E-3A airborne early warning systems; in advanced fighters (F-14, F-15, F-18) and interdiction aircraft (A-6) for all weather bombing; and in stand-off target acquisition systems (SOTAS).

• MISSILE GUIDANCE SATELLITES - Processors of radar and infrared sensor data for inertial navigation (Global Position Satellite), and processors for target recognition, proximity fusing, and clutter rejection in air-to-air missiles (Phoenix, Sparrow, Sidewinder), surface-to-air missiles (Patriot, Hawk), and submarine-launched cruise missiles.

° COMMUNICATIONS - Spread spectrum and time dispersion modulators and demodulators, error correction coders and decoders for digital voice transmission (ANDVT) and battlefield communications (REMBASS, Seek Talk, SINGGARS).

• SIGNAL INTERCEPT ANALYSIS- Signal modulation analyzers and signal classifiers for scan receivers (ALR-66, ALR-67).

° ELECTRO-OPTICAL PROCESSORS - Processors of electrooptical data for more detail and for estimation of target trajectories, in such infrared surveillance systems as the Halo satellite.

The microcomputer has basically the same capabilities and limitations as any other computer. These are:

- 1. Speed: Extremely rapid rates.
- 2. <u>Flexibility</u>: May be programmed to solve many types of problems.
- 3. <u>Repetitive Operation</u>: Perform similar operations thousands of times, without becoming bored, tired, or careless.
- 4. Accuracy: As specified by the programmer.
- 5. <u>Intuition</u>: Has no intuition. A man may suddenly find the answer to a problem without working on details, but a computer can only proceed as ordered [4].

The microcomputer offers a number of advantages to the design and production engineer, namely: 1) smaller size and weight, 2) greater reliability and flexibility, 3) component standardization, 4) shorter design cycle time, and 5) lower cost.

Economies associated with the "computer-on-a-chip" have resulted in the availability of systems with the functionality and performance of larger computer systems at a cost which is up to two orders of magnitude smaller.

Multi-microcomputer systems are presently designed and will be in a large number of applications, such as control of electric power, nuclear power generating facilities, etc.

The reasons why these systems are useful in many applications are several. Figure 6 tabulates the advantages and disadvantages of multi-microcomputer systems [23].

a state of the state of the

	ADVANTAGES		DISADVANTAGES
1.	Increased reliability	1.	Increased software complexity
2.	Increased survivability	2.	More dependence on com-
3.	Increased distributed processing power		munications technology
4.	Increased responsive- ness	3.	Unique expertise needed during design and development phase
5.	Increased modularity		
6.	System expandability in smaller increments		
7.	Lower cost		

Figure 6. Advantages and disadvantages of multimicrocomputer systems.

The March 1979 Quarterly Report for the Chemical Fund, Inc. states that: "Many analysts foresee a proliferation of intelligent electronics so widespread over the next ten years that the 1980's may be known as the Microcomputer Decade."

It is estimated that the annual demand for microcomputers may reach 300 million units by 1984, and triple that number may be in use by the end of this decade.

The following table I and figure 7 present the characteristics of different computer systems and corresponding execution times.

General Characteristics	IBM 370/168	DEC PDP 11/45	UN IVAC AN/UYK-7	UNIVAC AN/UYK-20	ZILOG Z-8000*
Туре	Maxi	Mini	Maxi	Mini	Micro
Construction Standard	Comm.	Comm.	Mil.	Mil.	Comm.
Word Length (in bits)	32	16	32	16/32	16
Processor Add Time (in µS)	0.13	0.90	1.50	0.75	0.75
Maximum I/O Data Rate (bytes/sec)	16M	9M	4M	65K	48M
Number of Internal Registers	64	16	8(16)	16(32)	16
Maximum Physical Dimensions (in inches)	Huge	71x30x22	41x24x20	24x18x29	14x2x11
Maximum Weight (in 1bs)	out of range	300	527 to 1,139	185	2
Multi-User Op. System	Yes	Yes	Yes	No	Yes
Real-Time O.S.	Yes	Yes	Yes	Yes	Yes
Security	Yes	Yes	Yes	No	No
Memory Capacity (in words)	8.4M	248 K	256 K	65 K	8 M
Software	All Types	Wide Variety	Fortran, CMS-2 Assembly	CMS-2 Assembly	Basic, Cobol, Fortran, Assembly
Approximate Cost	4.5M	50 K	250 K	80 K	1300

# Different Types of Computers

K = Thousands, M = Millions, Comm. = Commercial, Mil. = Military

\*This information is for a single-board computer (SBC) and not for a complete microcomputer system.



#### III. THE USE AND PERFORMANCE REQUIREMENTS FOR COMPUTER SYSTEMS IN TACTICAL APPLICATIONS

#### A. GENERAL VIEW

The use of computers in tactical applications in the Navy really got off the ground around 1960, with the fleet ballistic missile submarines (SSBN) and the Naval Tactical Data System (NTDS).

The submarine's missile and navigation systems contained several computers, and the NTDS was a complete automation of the ship's Combat Information Center (CIC). The Airborne Tactical Data System (ATDS), an airborne extension of NTDS, followed soon after [10].

Some of the most well known computer systems which are in use today for tactical applications are MAXI and MINI, as IBM 370/168, AN/UYK-7, AN-UYK-14, AN/UYK-20, etc. Although the microcomputer systems are not used for tactical applications, the use of microprocessors as integral components of computer's peripherals, in order to make them more intelligent (i.e., RD-358 magnetic tape, display consoles, etc.) is virtually impossible to enumerate.

The complete spectrum of current tactical applications will be surveyed starting with those of sensor control and ending with those of guided missile control.

#### **B.** SENSOR CONTROL

Radar, sonar and electronic warfare devices are the prime examples. These analog devices use computers for converting incoming signals from analog to digital form, and for automatic detection and classification of targets, track correlation and automatic tracking. Without a computer, a typical real-time electronic warfare system depends on the operator's ability to spot significant signals, to analyze and pass the information to CIC. Using the Automatic Detection and Tracking (ADT) radar system, it is possible to track simultaneously hundreds of targets, whereas a human operator is only able to handle up to four targets.

#### 1. Single-Function, Stand-Alone Systems

Most familiar and widely used systems are the navigations systems, such as Satellite, Inertial, OMEGA, LORAN, TACAN, and autopilots. The F-14 uses a computer to control its variable wing angle and the A-7 has an automated in-flight engine monitor system.

C. COMMAND AND DISPLAY

In the area of operations, the most well known computerized systems are the Command and Control (or real-time combat direction) systems. Such systems are the NTDS, ATDS, Marine Tactical Data System (MTDS) and the AEGIS Combat System, detailed in figures 8, 9.

These systems consist basically of the following major components:


Contraction of



35

· · · · · · · · ·

## 1. Analog to Digital Converters

These are used in order to convert the information from the sensors (radar, sonar, EW/ESM, etc.) into digital form and enter directly into the digital computer.

#### 2. Computing Equipment

The heart of these systems is a central computer system(s) complex with various man/machine interface devices, in order to perform necessary calculations, in accordance with stored programs, and provide a picture of the current situation to the operator for further exploitation.

#### 3. Communications Equipment

The crucial need of exchanging information rapidly between the elements of a unit or between different forces, requires the use of high-speed data-links. Via these datalinks, the computers can communicate very fast (by having the appropriate data) in order to provide the necessary information for the operations.

#### 4. Communications

Digital computers are excellent communication tools. They can interface directly with dedicated equipment; they can detect and correct errors in the messages; they can be used as controllers and multiplexers for communicating asynchronously through several devices simultaneously. Some examples are:

- a) Ship-to-shore communication
- b) Ship-to-ship communication
- c) Inter-computer data links

- d) Automatic and semi-automatic aircraft control links
- e) Missile and drone guidance
- f) Telemetry
- g) Broadcasting tactical information to teletypes.

#### 5. Visual Displays

The human interface for the computed information is the visual console display, where the operator finally has to make his own decisions.

6. Test Equipment

It may sound unusual to use the words "test equipment" in a paper about tactical applications, but it is absolutely necessary to examine this type of equipment because without it, it is not feasible to verify the operational readiness of the equipment used in tactical applications.

The automatic test equipment systems can be made to test anything, from tanks to turbines and to total aircraft systems. A notable example is the "Versatile Avionics Shop Test" (VAST) which determines the status of the system of an aircraft and what maintenance is needed when a plane returns to the aircraft-carrier. The microcomputers in these systems will make them more powerful, faster and cheaper.

#### D. WEAPONS CONTROL

This type of system obviously must be capable of standing alone, and is a natural application for automatic control by a computer.

Modern naval warfare requires storage of large amounts of information concerning the location, movement, and characteristics of friendly and enemy units. The air warfare problem requires that fire control computers be almost fully automatic with little need for manual operation.

Today, all new combat ships and aircraft are being built with digital fire control systems, and original equipment on most older units is being retrofitted with digital computers. U.S. Navy has adopted two computers to be standard naval ship tactical digital processors, AN/UYK-7 which can be configured for either small or large fire control system applications, and AN/UYK-20 which is designed to meet most of the small and medium-sized needs.

In the area of weapons control, typical tasks performed by today's digital computers include director positioning, automatic target acquisition and tracking, computation of all gun or missile orders, weapons selection, weapons firing, and control of operator's display consoles. It is believed that in the near future all these tasks will be solved by micro or multi-microcomputer systems.

## E. GUN CONTROL

This system is responsible for acquiring, locking onto and then tracking the target; solving the ballistics control problem continuously, and finally, guiding the weapon's bearing and elevation, after receiving ship's data concerning course, speed, roll-pitch and wind speed and direction.

Coupled with a scanning radar, a digital computer or microcomputer system can be used to allow the radar to track selected targets while searching for new contacts. By this way, several different contacts can be simultaneously engaged.

F. GUIDED MISSILE CONTROL

In the AEGIS system tests conducted in 1975, two standard missile firings destroyed two low altitude supersonic missile targets simulating attacks on an aircraft and an aircraft carrier. Each target was shot down with only one human action --the closing of the firing key [10].

Digital computers have been combined with phased-array radar. In this radar, the transmitted radar beam is steered in bearing and elevation by varying the phases of the signals transmitted by each of the elements on each antenna face. This radar and the digital computer system associated with it can do all of the following automatically:

- ° Track-while-scan.
- ° Conduct a continuous 360-degree search.
- Transmit mid course guidance commands to several missiles in flight, each missile being aimed at a different target. When the missile acquires the target terminal guidance by the missiles own sensors takes over.
- \* Automatically initiates countermeasures when jamming is detected.

The combination of all the above features in the digital control system gives the missiles capability to achieve far greater ranges than their predecessors.

#### G. A CRITICAL QUESTION ABOUT MICROCOMPUTERS

The previous paragraphs examined the use of the computer systems in general tactical applications. At this point the question is:

"Would it be possible, in the near future, to use computers which are similar to today's commercial microcomputers for tactical applications?"

It is believed that even the most pessimistic expert on computer technology will give an affirmative answer, because of the following reasons:

1. The use of the  $\mu C$  in areas of science and engineering similar to tactical applications increased rapidly in the last five years, and much more increase is expected in the future.

2. The general capabilities of the newest  $\mu C$  are approaching the minicomputer and in some areas exceed those of mini computers.

- 3. The dimensions and the weight of the  $\mu C$  are very small.

4. The possibility of using many (up to 8 recently and 16 in the near future) single-board  $\mu$ Cs in parallel (multimicrocomputer systems) in order to achieve the same results as a single maxi or a large mini computer. The advantages are: reducing the cost approximately 10 to 1 and increasing the capabilities of weapons (i.e., surface-to-surface short range missiles).

5. The low hardware cost of any  $\mu C$  system in comparison with mini or maxi computers; and

6. The need for increased effectiveness of weapon systems in order to remain competitive or become superior to adversaries.

For all these reasons above, it is believed that very soon the  $\mu$ C systems will take their proper place in the Armed Forces and in this way even the smallest combat unit can afford a powerful new tool.

#### IV. METHOD OF EVALUATION

The major questions to be answered when evaluating microcomputer systems for a particular application are:

- 1. Will the system have enough computing capacity to satisfy the computational requirements?
- 2. For the systems which have enough computing capacity, will the systems have support software, pass military qualifications, have multiple sources, etc.?

Question 1 above is addressed in two parts:

Computing capacity of uniprocessors.

Computing capacity of multi-microcomputer systems.

Question 2 above is developed in two parts:

Additional important criteria.

Desirable features.

#### A. COMPUTING CAPACITY OF UNIPROCESSORS

The measures of capacity which significantly limit the computational capacity are:

- \* Read only memory (ROM) capacity.
- ° Random access memory (RAM) capacity.
- <sup>o</sup> Auxiliary memory capacity, disks, tapes, etc.

The memory capacity limits have played an important role in the past because the original requirements of the problem changed and caused an overflow which in turn caused the use of more unicomputers or a revision of the original program to fit into the existing memory. In all cases, substantial unanticipated expenditures were required to adjust the system to the new requirements.

With microcomputer systems, the memory size limits range from 1-2 Kbytes, on single chip computers, to 65 Kbytes on single board computers, to 16 Mbytes in the newest 16-bit systems. Tables VI, IX, X, and XI tabulate the implemented memory sizes for typical microcomputer systems.

A widely used performance capacity measure is Million Instructions Per Second, MIPS. This measure is misleading because of the wide range of instruction execution times with microcomputers. At least four different categories of instructions must be distinguished to give the crudest measures:

° INTEGER Additions, Subtractions and Comparisons.

- ° INTEGER Multiplications.
- ° FLOATING Point Additions and Comparisons.
- ° FLOATING Point Multiplications and Divisions.

The case study in the following chapter distinguishes between twelve basic functions and thus gives a more refined evaluation of a system.

The performance capacity of a system to carry out required calculations depends on the mix of calculations. The commonly used mixes are: Gibson, Real-Time, Message Processing (table IIa) [29].

The attached Basic program was written in order to calculate performance capacities of representative computer systems, standard Military computers and widely used microcomputer

<b>FABLE IIa</b>
------------------

and the second

Performance Capacities of Different Computer Systems. Three commonly used mixes are summarized below.

System Mix	Gibson	Real-Time	Message Processing
Fixed (single prec.)			
Add/Subtract	0.061	0.16	0.05
Multiply	0.006	0.05	0.005
Divide	0.002	0.02	0.005
Logical			
Compare	0.038	0.12	0.01
Shift (6 bits)	0.044	0.05	0.03
And/Or	0.016	0.04	0.15
Control			
Load/Store	0.312	0.33	0.47
Conditional branch	0.166	0.1	0.14
Increment and store index	0.18	0.04	0.03
Move reg. to reg. No memory refer.	0.053	0.05	0.058
I/O Control			
Programmed I/O transfer	0	0.02	0
Initialize buffered I/O	0	0.01	0.001
Interrupt response and store 4 reg.	0.122	0.01	0.001

systems, using three common instruction mixes (Gibson, Real-Time and Message Processing). Table IIb contains the instruction times (in  $\mu$ s) for these computers.

The results of these calculations are given in table IIc. In order to answer the question, whether or not a given computer system has the performance capacity to satisfy the requirements of an application, one must characterize the application in terms of the instruction mix and the number of instructions in this mix to be executed each second. If the capacity of the computer exceeds the requirements, the computer system can be used to carry out the calculations. If the capacity of the computer is smaller than the stated requirements, then either a faster computer must be found or a multicomputer system must be used.

## B. COMPUTING CAPACITY OF MULTI-MICROCOMPUTER SYSTEMS

The computing capacity of a uniprocessor system is a serious limitation for applications which at the design stage fit comfortably into a uniprocessor, but several years later outgrow the capacity of uniprocessor. Historically this has been the case with tactical systems, such as NTDS. Even for the newest tactical system, AEGIS, the number of computers in the system has been growing.

A uniprocessor system designed during 1960-70 was not designed to be a building block of a multicomputer system. The microcomputer systems, on the other hand, were specifically designed to be building blocks of larger systems.

TA	BL	E	Ι	II	b

Description of the second second second

Performance	Capacit	ties of	E Diff	ferent Co	omputer	Systems
Instruction	times	(in µs)	) for	differe	nt compu	iters.

Instruction	AN/UYK 7	AN/UYK 20	AN/UYK 14	MC 6800 (:1MHZ)	INTEL 8086 (SMHZ)	ZILOG 8000 (4MHZ)	MC 68000 (8MHZ)
Fixed (s.p.)							
ADD/SUB reg. to register	1.5	1.5	2.0	5.0	0.6	1.0	0.5
MUL (16 bits)	7.5	4.2	5.3	72.0	26.0	18.75	5.4
DIV (16 bits)	14.5	7.2	10.6	98.0	32.0	23.7	10.9
Logical							
COMPARE (reg. immediate)	1.1	1.2	2.0	2.0	3.0	1.75	1.4
SHIFT (6 bits, register)	1.75	3.5	2.7	12.0	2.4	1.75	1.3
AND/OR (reg. imm.)	1.0	1.5	2.0	2.0	3.0	1.75	1.5
Control							
LOAD/STORE	1.5	1.5	0.9	4.0	2.8	1.75	1.5
COND. BRANCH	1.5	2.25	1.75	3.0	1.6	2.75	0.6
INCR. & STORE	1.5	2.24	1.4	4.0	4.0	5.0	1.3
MOVE register to register	1.1	3.6	7.4	2.0	0.4	0.75	0.25
I/O Control							
Programmed I/O transfer	3.5	4.5	3.5	4.0	2.0	2.5	0.5
Initialize buffered I/O	3.25	4.5	5.8	4.0	4.0	4.0	4.0
Interrupt response & store 4 reg.	<b>e</b> 4.0	4.0	4.2	18.0	21.2	9.0	4.9

Note: Some given times are coming from separate calculations (MUL and DIV for MC6800), despite lack of such instructions in these computers.

```
10
        REM ** PROGRAM WRITTEN IN TRS-80 DISK BASIC.
20
        REN ** THIS PROGRAM CALCULATES PERFORMANCE CAPACITIES OF
30
        REN ** REPRESENTATIVE COMPUTER SYSTEMS USING THREE COMMON
40
        REM ## INSTRUCTION MIXES. THESE MIXES ARE GIBSON, REAL-
        REN ++ TIME, AND MESSAGE PROCESSING.
50
68
        CLEAR 5000
79
        DIM B(8, 13), G(8, 13), R(8, 13), M(8, 13)
        DIM N(20), P(8, 3), C(3, 20)
80
        INPUT DO YOU WISH TO CREATE A NEW FILE (Y/N)"; A$
85
86
        IF R$ = "N" THEN 670
90
        REM ** INPUT COMPUTER CHARACTERISTICS
100
        INPUT HOW MANY SYSTEMS DO YOU WISH TO EVALUATE ?"; N
        INPUT"ENTER NUMBER OF CONSTRATS"; M
105
110
        FOR I = 1 TO N
128
          INPUT"ENTER SYSTEM NAME"; N$(I)
          FOR J = 1 TO M
140
            PRINT"ENTER SYSTEM CONSTANT "J
150
155
            INPUT B(I, J)
160
          NEXT J
170
        NEXT I
        REM ** INPUT INSTRUCTION MIX CONSTRUCTS
189
        INPUT"DO YOU WISH TO ENTER NEW MIX CONSTRATS (Y/N)"; A$
181
182
        1F A$ = "N" THEN 790
198
        FOR I = 1 TO 3
200
        IF I = 1 THEN PRINT"ENTER GIBSON CONSTRATS"
        IF I = 2 THEN PRINT"ENTER REAL-TIME CONSTANTS"
210
        IF I = 3 THEN PRINT"ENTER MESSAGE PROCESSING CONSTANTS"
220
230
          FOR J = 1 TO M
240
            PRINT"ENTER EVALUATION CONSTANT "J
258
            INPUT C(LJ)
260
          NEXT J
265
        REN ** STORE INSTRUCTION MIX DATA TO DISK
278
        NEXT I
271
        OPEN"O", 1, "MIXDATA/TXT"
272
        PRINT#1, M
273
        FOR K = 1 TO 3
274
          FOR J = 1 TO M
275
            PRINT#L C(K, J)
276
          NEXT J
277
        NEXT K
```

```
278
        CLOSE 1
280
        REM ** PERFORM CALCULATIONS
290
        FOR I = 1 TO N
300
          FOR J = 1 TO M
310
            G(I, J) = B(I, J) + C(1, J)
320
            R(I, J) = B(I, J) + C(2, J)
338
            M(I, J) = B(I, J) + C(3, J)
340
          NEXT J
        NEXT 1
350
370
        FOR I = 1 TO N
388
          G1 = 0: R1 = 0: M1 = 0
398
          FOR J = 1 TO M
            G1 = G(1, J) + G1
400
418
            R1 = R(I,J) + R1
428
            M = H(LJ) + M
          NEXT J
430
448
          P(1,1) = G1
450
          P(1,2) = R1
460
          P(1,3) = M1
470
        NEXT I
        REM ** PRINT OUTPUT
475
        LPRINT TAB(35)*TABLE*
489
485
        LPRINT: LPRINT: LPRINT
490
        LPRINT TAB(13) "PERFORMANCE CAPACITIES OF DIFFERENT
        COMPUTER SYSTEMS"
495
        LPRINT: LPRINT: LPRINT: LPRINT
508
        LPRINT TAB(13) "SYSTEM" TAB(29) "GIBSON" TAB(40)
        "REAL-TIME"TAB(53) "MSG. PROCESSING"
510
        LPRINT TAB(13)
10
        :LPRINT:LPRINT
520
        FOR I = 1 TO N
530
          LPRINT TAB(13)N$(I)TAB(29)
          P(1,1)TRB(41)P(1,2)TRB(56)P(1,3)
535
          LPRINT
549
        NEXT I
545
        INPUT"DO YOU WISH TO SAVE DATA TO DISK (Y/N)"; A$
546
        IF A$ = "N" THEN 1000
550
        REM # STORE DATA IN A DISK FILE
568
        OPEN"0", 1, "PERFDATA/TXT
565
        PRINT#L N. M
```

578 FOR I = 1 TO N 589 PRINT#1, N\$(I) 598 FOR J = 1 TO H 690 PRINT#1, G(I, J), R(I, J), M(I, J) 610 NEXT J 620 FOR K = 1 TO 3 630 PRINT#1 P(1,K) 640 NEXT K 650 NEXT I 668 CLOSE 1 665 GOTO 1000 666 REM \*\* INPUT SYSTEM PERFORMANCE CHARACTERISTICS 670 OPEN"I", 1, "PERFDATA/TXT" 675 INPUT#1, N. M 688 FOR I = 1 TO N 690 INPUT#L N\$(I) 700 FOR J = 1 TO M 710 INPUT#1, G(I, J), R(I, J), M(I, J) 728 NEXT J FOR K = 1 TO 3 730 740 INPUTHL P(L,K) 750 NEXT K 760 NEXT I 778 CLOSE 1 789 GOT0 360 785 REM 👐 INPUT INSTRUCTION MIX DATA FROM DISK 790 OPEN"I", 1, "MIXDRTR/TXT" 800 INPUT#1. M 810 FOR K= 1 TO 3 . ... 820 FOR J = 1 TO M INPUT#1, C(K, J) 839 840 NEXT J 850 NEXT K 868 CLOSE 1 878 GOTO 288 / 1000 END

#### TABLE II c

PERFORMANCE CAPACITIES OF DIFFERENT COMPUTER SYSTEMS

SYSTEM	GIBSON	REAL-TIME	MSG. PROCESSING
AN/UYK-7	1. 8336	2. 067	1. 42955
AN/UYK-20	2. 2782	2. 1376	1. 7785
AN/UYK-14	2. 1297	2. 32	1. 7297
MC-6800	6. 337	9. 46	4. 338
INTEL-8086	4. 991	4. 192	2. 5804
Z-8000	3. 43265	3. 209	2. 00875
MC-68000	1. 63175	1. 5795	1. 2359

Note: The number in the above table refers to the number of instruction mix per microsecond.

A typical multi-microcomputer system consists of single board computers, each of which contains a processor (Proc), input/ output interfaces (I/O), real-time clocks (RTC), interrupt control (IC) and private memory. The single board computers are connected by a system's bus, which allows common shared memory to be used for passing information between such system almost at the same rate as a computer accesses its private memory. Typically, up to sixteen such single board computers

can be put into one system. A representative multimicrocomputer system is shown in figure 10.



Figure 10. A multi-microcomputer system.

The computational capacity of such systems depends on the single board processors.

Each processor adds memory capacity as well as processing capacity to the system. If the processes can be partitioned among the processors in such a way that each processor is working near its performance capacity and relatively little information is shared between the processors, then n processors yield a performance capacity of n times uniprocessor performance capacity.

51

the state of the

If, on the other hand, much information needs to be shared, the system's bus may become a serious bottleneck to the performance capacity. Experimental evidence indicates that if the system's bus is used at 50% or less of its capacity, the performance of each single board processor is not affected. The system's bus usage at 75% capacity caused a loss in performance capacity of about 10% in the system. Use of the bus beyond 80% capacity causes serious losses in performance of the system.

In order to predict the system's bus usage, the number of external variables stored in common memory and the frequency of updates must be estimated. Each time the variable is used by another process resident in another single board computer, another access of the system's bus is necessary. Therefore, the bus use estimate is based on the number of variables multiplied by the frequency of updates and usages multiplied by the bus transaction time of the common memory reference.

In the case study in the following chapter, the system's bus usage is a minor percentage of bus capacity and hence the performance capacity is a multiple of the performance capacities of the individual computers in the system.

Because the performance of the multi-microcomputer system depends on the partitioning of the processes into relatively independent partition elements, the performance capacity is dependent on the application and therefore must be studied in the context of the application. If the bus usage remains less than 50% of bus capacity, the performance capacity is the sum

of the performance capacities of the single board computers in the system.

## C. ADDITIONAL IMPORTANT CRITERIA

1. Military Qualifications

a. Dimensions

As it is known, there are a lot of restrictions in warships, submarines, and especially in aircraft and helicopters about additional equipment, because of the restricted space in these units. Some examples of these restrictions are given below:

- (1) The door openings in surface ships are  $26 \times 60$  inches.
- (2) The diameter of hatch openings in submarines is 21 inches.

Therefore, the smaller the size the more acceptable.

b. Weight

Another important factor which must be very carefully examined is weight, because for any equipment destined for fast boats, aircraft, helicopters, spacecraft, and satellites, equipment must be as small as possible. It must be kept in mind that for an extra pound of weight in an aircraft and similar vehicles, they must add seven pounds weight of additional material to its construction [30]; that means the ratio of extra weight in these cases is seven to one. On the other hand, speaking about fast boats, it must be kept in mind that extra weight on these has a drastic influence on reducing their highest speed.

### c. Electrical Power

To operate the microprocessors in most of the new systems, requires a five (5) to twenty (20) volt power supply. However, many of these power supplies are not regulated and ships are famous for having fluctuations, and part or total loss of electrical power. The necessary prerequisites are that the system has a regulated power supply that can withstand fluctuations of 10%, based on the maximum fluctuations that most shipboard electronic equipment must be able to withstand without degradation, and also has a back-up battery with automatic cut-in as power fails. If the system does not have one of those prerequisites, it is not considered capable for shipboard use.

d. Environmental Conditions

These are:

- (1) <u>Temperature</u>: It has to withstand temperature ranges of 32 to 100 degrees Farenheit to remain in contention.
- (2) <u>Humidity</u>: It has to be operable under relative humidity between 10% and 90%.
- (3) <u>Shock Mounts</u>: It has to provide shock mounts for the units.

In order for a microcomputer to be able to withstand these extremes of temperature and humidity, power supply cooling features have to be incorporated into the computer.

2. Reliability

a. Time of Failure/Repair

The minimum "mean Time Between Failures" (MfBF) of 3,000 hours is required for a unit to remain in contention and receive further evaluation. The maximum of 60 minutes is allowed for "Mean Time To Repair" (MTTR), utilizing the manufacturer's recommended spare parts allowance. It should not be contingent on any extraordinary technical training either.

b. Manufacturers

To determine the best manufacturers of the major components of the system, such as microprocessors and peripherals, is not an easy job, especially when a company is new and small. Therefore, the size of the company, and how long it has been in this kind of business is a very important factor.

3. Maintainability

One of the most important factors to be examined is the availability of replacement parts when a failure occurs to the system. This factor, it can be said, is more important than that of the time between failures.

To simplify maintenance, a diagnostic package, containing the following, must be included:

a. Front panel indications

b. Test points

c. Test programs

d. Test equipment.

In the area of chips that are used, it is more reliable to use the expensive chips that are normally soldered on the boards than those that plug-in.

In order for the system to remain continuously updated, it must be included in maintenance support, which is

55

A COMMON

the manufacturer's obligation for any software or hardware changes to the system.

If a training program for the maintenance and operator personnel is available, that will be counted as an advantage.

4. Programming Languages for Microprocessors

Programs can be written at various levels. Machine and assembly language do not require much software or hardware support and can produce very efficient programs. These languages, however, are machine-dependent and difficult and time-consuming for the programmer to use. High-level or procedure-oriented languages are not machine-dependent and are significantly easier to use but require a large amount of software and hardware support and usually produce inefficient programs (figure 11).

No less than twenty different microprocessor languages are in use today. Languages can be classified in several ways, considering the use for which they are best suited. Most  $\mu$ C applications seem to be in the area of system programming or control applications (i.e., they require direct access to the machine to perform I/O or control functions or to achieve maximum efficiency). The primary dimension for classifying a language is its distance from assembly language (i.e., its level). There are four categories into which languages are divided: structured assemblers, high-level machine-dependent, high-level  $\mu$ P-oriented and high-level system languages.



Fig. 11. A genealogy of current languages for microprocessors. A link indicates that the design philosophy of the descendant was strongly influenced by the ancestor but does not imply that one contains the other.

(From: "A Survey of Microprocessors Languages," by S. C. Reghizzi, <u>Computer</u>, January 1930.)

## a. Structured Assemblers

The main impetus to the introduction of High Level Languages (HLL) arose from the awkwardness and poor readability of assembler programs, shortcomings which led to high development costs and difficulties in updating programs.

On the other hand, the disadvantage with HLL's is that they take more memory space and more time to execute than assembly languages. A lengthy program in HLL can take an hour to compile. The additional execution time is a disadvantage in some real-time applications and the added memory requirements become critical in single-chip computers with limited read-only memory [13].

A modern structured assembler is PLZ-ASM, a language designed by Zilog to interface with higher-level language PLZ-SYS.

b. High-Level Machine-Dependent Languages

These languages are less machine-dependent than structured assemblers, since executive statements and expressions are written in conventional high-level notations rather than in assembly form. The ancestor of these languages is PL 360, designed by Wirth in 1968 for the IBM 360. To understand a PL language, one has to have a pretty good idea of the machine instructions and machine-idiosyncratic conventions.

In this category four languages are available for microcomputers by increasing levels: Smal/80, BSAL/80, Mistral and PL/65.

58

c. High-Level Microprocessor-Oriented Languages

These languages--that are largely machine-independent--are especially designed for  $\mu P$ . They avoid some of the resource-consuming features of traditional main-frame languages which would cause unacceptable performance degradation.

In this category are: MPL, PLuS, PL/M and PLZ-SYS. Information and characteristics of these languages are given in table III.

d. Higher-Level System Languages

Excluding APL, Cobol and Fortran (as non-system languages), four representative members of this family are: Basic, Pascal, RTL/2 and C, all originally designed for larger machines.

"BASIC" is available on all microsystems, although some Basic implementation have low quality. Several extensions and dialects derived from Basic are currently available on microcomputers (e.g., LLL Basic, which facilitates interfacing with assembly routines and has been successfully used in a real-time environment).

"PASCAL" is perhaps the best choice for achieving machine-independence on 16-bit (but not on 8-bit) machines in noncritical applications. Pascal will likely be available on most new generation machines; this will be facilitated by the general availability of a portable Pascal compiler which can be easily boot-strapped on new machines. The University of California at San Diego has boot-strapped Pascal (or rather a

# TABLE III

# Information and Characteristics of Languages--MPL, PL/M, PLZ-SYS

1.	<b>General Information</b>	on MPL.	PLuS.	PL/M.	and PLZ	SYS.
_						

LANGUAGE	MPL	PLUS	PL/M	PLZ-SYS
ORIGINALLY DESIGNED FOR	Motorola 6#00	Skynetkus 2650	intel 8080	Zung Z-80
YEAR	1976	19/4	1975	1977
NOW AVAILABLE OR ANNOUNCED FOR	M68000 M68000		inter Allah MGBOI (trom Independent Su <b>pplier</b> )	Ziling Z-8000 Obvetti
CROSS COMPILER	tor M6800	tar 2650	tar 8080	Under de velapis <b>ent</b> by non Zilog suppliers
RESIDENT	Motorola 6800	•	Jintel 8080 8085 8085	Zilog Z-80 Olivetti maclumes
RESIDENT	-		•	Zing 2-80

LÂNGUAGE	MINIMAL MEMORY CONF GURATION REQUIRED (RBYTES)
PLZ-ASM	35
Smail/80 Mistral M6800	12* 8*
PL65	16
MPL	56
PL/M 8090	64
PL/M 8085	64
PLZ-SYS	48
Basic Zilog	48
Basic Intel 8080	48
LLL Basic	12*
Pascal UCSD 8080	48*
Pascal Texas Instr.	. 48

### C. Simple data types in MPL, PLuS, PL/M, and PLZ-SYS.

LANGUAGE	811	1-BYTE BINARY	CHAHACTER	WORD (2 BYTES BINARY)	POINTER (2 BYTES)	1 BYTE SIGNED INTEGER (BINARY)	2-BYTES SIGNED INTEGER (BINARY)	SIGNED OR UNSIGNED ASCII CODED DECIMAL	BOOLEAN OR LOGICAL	LABEL
MPL	yeti	yna	yes	y85	same as 2 bytes binary	-	-	yes, up 19 12 digits	only unnamed Boolean expressions in if and de while statements	yes, label dec- larations
Rus	-	yes	785	same as address	same as adgress	yina	yes	-	1-byte ur 2-byte types can be inter- preted as true or false under standard conventients	yms
Pt / M	-	yes	same as 1-byte binary	same as address	same as address	-	-	-	1-byte or 2-byte types can be inter- preted as true or faise under standard conventiens	yes, labet doctara- lions
m.Z-SYS	-	yes	same as 1-byte binary	y <b>es</b>	y <b>es</b>	yes	<b>795</b>	-	only unnamed Boolean en- pressions in if statements	yes. resincted to de groups



d. A spectrum of language processing solutions from full translation to source interpretation.

	2·ways	TESTS Multi-ways ( <i>case</i> -statement)	Interations	go to F	rocedure calls and returns	Halt
MPL	it then, it then else	Computed goto (as in Fortran).	do + = initial to tinal by increment do while Boolean-exp Parameters restricted to constants or variables.	yes	yes	-
PLu <b>S</b>	ıf ∴.then, ıtthen eise	One out of n branches is selected according to the value of a control expression in the range $0 \dots n - 1$ .	do i = initial to final by increment; Parameters may be expressions. do while expression do until expression	yes	yes	yes
<b>PL/</b> M	ifthen, ifthen eise	Same as PLuS.	Same as PLuS, except that there is no <i>do until</i> and increment>0.	yes	yes	yes
PLZ-SYS	<i>il then</i> fi it then eise fi,	The i-th branch is entered if a control expression equals one of $K_{11}$ , $K_{12}$ . $K_{m}$ (a list of constants), for some $1 \le i \le n$ . Otherwise the (n + 1)-th branch is entered	Never ending loop; do	Restricted to: exit [from label] repeat [from label] where label denotes a loop.	yes	-

#### Control structures in MPL, PLuS, PL/M, and PLZ-SYS.

-

61 '

## TABLE III (continued)

	Allowed actual parameters	Parameter transmission	Call by address of 1st instruction	interrupt procedures	Reentrant (recursive) procedures	Functions	Multiple exits	Muitiple return values
MPL	Constants, nonsubscripted variables, array names Parameters are forbidden as control variables in control statements, subscripts, and parameters of an inner procedure.	By value. (by address for arrays and based variables). User specifies transmission via registers.	-		no	no		y <del>es</del>
PLuS	Expressions, addresses of arrays. Automatic conversion of actual parameters types to formal parameters.	By value (by address for based variables). The last two parzmeters and result are passed via registers	-	-	no	yes	yes	-
PL/M	Expressions, addresses of arrays or structures	Same as PLuS, but parameters via stack for reentrant procedures	<del>ye</del> s	yes	Must be explicitly declared	yes	yes	-
PLZ: SYS	Constants and variables. Pointers to arrays and structures.	By value (by address for pointed variables)			Mutually recursive procedures must be declared in separate modules	yes	no	yes

#### f. Procedures and parameter passing in MPL, PLuS, PL/M, and PLZ-SYS.

g. Comparison of languages processing solutions with respect to program execution, program preparation, and portability. Numbers denote qualitative ordering. Note that the evaluation is very much dependent on the source language and on the target machine.

		COMPLETE TRANSLATION	TRANSLATION TO ASSI.MBLER	TRANSLATION TO AMC AND AMC INTERPRETATION	SOURCE LANGUAGE IN TERPRETATION
program preparation program execution	EXECUTION SPEED	1	1	2	3
	SIZE OF UBJECT PROGRAM	3	3	2	1
	SIZE OF RUN-TIME SYSTEM	1	١	2	3
	RUN-TIME DIAGNOSTIC AND DEBUGGING	3	3	2	1
	OVERALL SIZE OF PROGRAM PROCESSING PACKAGE	3	4 (include\$ assembler)	2	1
	TIME TO PREPARE A PROGRAM FOR EXECUTION	3	4	2	1
	DIAGNOSTICS AND	2	3	j	1
	MEMORY SIZE TO COMPILE	ف	2	1	-
porability	PORTABILITY TO OTHER MACHINE	3 (good with retargeting techniques)	2	1	2

(From: "A Survey of µP Languages," S. C. Reghizzi, Computer, Jan 1980)

subset) on a number of different systems--i.e., Intel, Digital Equipment, Motorola, Texas Instruments and Zilog.

"RTL/2", originally designed by ICI, the British chemical firm for real-time control applications on microcomputers, currently is advertised as a microprocessor language. A Multi-Task System (MTS), mostly written in RTL/2, is available for mini computer DEC-PDP/11. Recently a micro-RTL/2 for Intel, Motorola and Zilog microcomputers was announced. These new compilers will use the time-honored technique of translation to an intermediate language, followed by interpretation (as with UCSD-Pascal) in order to achieve portability.

"C" is available on microcomputers and larger main-frames: a subset, "Tiny C", is available on several microcomputers as an interpreter. C can be compiled into reasonable code on a variety of machines (it was used to write the very successful operating system UNIX), but is not very readable due to the rich set of operators (more than 50), which can be combined in rather peculiar ways.

e. CMS-2 Programming Language

CMS-2 (Compiler Monitor System 2) is a computer programming system developed for the U.S. Navy. Primarily designed for real-time, command and control applications, it is equally applicable to other data processing and scientific problems.

CMS-2 was designed to replace the CS-1 compiler and MS-1 monitor system and developed in the late 1950's and early 1960's. CS-1 was used primarily for NTDS and is

oriented to the CP-642/USQ-20 family of computers, which have been replaced by newer computers, and is too machinedependent to meet new programming demands.

Within CMS-2 are five components (compiler, loader, librarian, utilities, and assembler) which are controlled by a routine called the Monitor. The standard input is punch cards and the standard output is source cards, binary cards or print lines.

Within a single job, several components of the CMS-2 system may be executed (i.e., compiling a source program for the AN/UYK-7 computer and loading the object code into the computer for execution).

CMS-2 system was designed in 1967 and from then on it is in use in the U.S. Navy as a basic language for tactical data systems. It is available to mini and maxi computer systems but not yet to microcomputer systems.

f. Ada Programming Language

Ada, which is named after the world's first computer programmer, Lady Ada Lovelace, is a very new programming language and is designed for the necessities of Department of Defense (DOD) by CII Honeywell Bull of France. The language has reached the fine-tuning stage, but the first productionquality compiler for the language probably will not be available until early 1982 [17].

The Ada compiler will consist of a root compiler that will produce machine-independent intermediate code and

object code generators that will translate the intermediate code into the machine language of specific computers. The Army has initially specified code generators for AN/GYK 12, PDP 11/70, VAX 11/780 and LITTON L3050. The Air Force asked for IBM 370, Perkin-Elmer Corp. model 8/32, DEC system 10, and the Control Data Corp. model 6600.

While the Army and the Air Force have embraced Ada, the Navy still has reservations [18]. INTEL Corporation reportedly is supporting Ada in its new microprocessor series [32].

Ada is an example of one potential microprocessor HOL standard, because it is being developed to serve programming needs for real-time processing, including microprocessor applications.

5. Operating System

The term "operating system" (0.S.) denotes those program modules within a computer system that govern the control of equipment resources such as processors, main storage, secondary storage, I/O devices, and files (figures 12, 13, 14). These modules resolve conflicts, attempt to optimize performance, and simplify the effective use of the system. They act as an interface between the user's programs and the physical computer hardware [7].

Sophisticated operating systems increase the efficiency and consequently decrease the cost of using a computer. At this point it is noted that, as estimated, 70% of the amount of money that is spent in a year on computer products and

65

· · · · · · · · · · · ·



درد باليدموريون يدره

þ

Figure 12 Relationship of O.S. to basic computer hardware.

a .....



services goes into software development, data processing salaries, and other software-related activities.

The O.S. is divided into two main categories: (1) developmental, and (2) target, or executive.

In the first category, they belong to O.S. which cover a lot of applications of many kinds of computer systems, and for that reason are very complex and their cost is high. "MULTICS" (Multiplexed Information and Computing Systems) belong to this category, which employs the concept of "rings of protection," links, hierarchical structure, and dynamic linking. Also, from March 1980, there is available, especially for microcomputers, the Taurus O.S. which can spool to multiple printers, has built-in security features with multiple priority levels and is capable of reporting the status of every running job in the system [15].

In the second category, the O.S. are very simple and are designed for a given computer to support specific applications (i.e., fire control system). The advantages of this O.S. are simplicity and low cost.

In order to evaluate the O.S. that are available for different microcomputers, it is necessary to determine what the general application area is (i.e., time-sharing, process control, etc.) and the efficiency of the O.S.

Transportability is also an important consideration relating to software cost and software development time.

The more programming languages and input and output devices that the operating system can support, the more powerful it is and the more desirable it becomes.

Obreat

The efficiency of an O.S. is also important. An O.S. may be able to support distributed systems or security, but if the necessary time to accomplish it is "long", i.e., 30 seconds delay waiting for feedback of an interactive system seems "long" and one minute seems "too long."

It is necessary to examine any of the following special features of an O.S.:

- a. Type of file protection or security provisions.
- b. Method of starting and restarting the system.
- c. Form of memory retention capability in case of power loss.
- d. Error recovery method such that the user will be able to understand them, without reference to publications.
- e. Ease of use.

D. DESIRABLE FEATURES

1. Availability

Availability (A) is defined as the percentage of time a computer system is available and is given by the following equation:

$$A = \frac{MTBF}{MTBF + MTTR}$$

For example, given:

MTBF = 3000 hours and MTTR = 48 hours

$$A = \frac{3000}{3000 + 48} = 0.98425 \simeq 98.4$$

MTBF = 3000 hours and MTTR = 1.5 hours
$$A = \frac{3000}{3000 + 1.5} = .99950 \simeq 99.9\%.$$

From these examples it can be seen that availability can be improved by increasing the MTBF and/or decreasing the MTTR.

2. Software Material

Software evaluation is so complex a task that it is at present more of an art than a science. The number of factors to be considered is so large and their interaction and implementation so complex that thorough evaluations of any but the simplest products are exceedingly costly and, as a result, are seldom if ever done [12].

Attention will have to be focused on the most important aspects of the evaluation process at the expense of the less important ones. A key question, then, is: "Which aspects are most important?" The answer hinges on two basic points:

(1) The software package <u>must</u> contribute in a nontrivial way to helping the user firm achieve its basic objectives, which generally include a certain type and level of service and long-term survival.

(2) The software package <u>should</u> be able to adapt to changes--most notable, growth of a company's volume of business--and continue to function in a useful way for as long as possible.

Nearly all objectives for software products fall into one or more of the following categories:

70

a Carlos

- (1) Functionality
- (2) Capacity
- (3) Flexibility
- (4) Usability
- (5) Reliability
- (6) Security
- (7) Performance
- (8) Serviceability
- (9) Ownership
- (10) Minimization of operating and maintenance costs
- (11) Minimization of purchase and installation costs.

The categories are certainly not mutually exclusive. For example, functionality, performance and capacity are often strongly interrelated. Some examples, by category, follow:

a. Functionality

Do the input transactions, files and reports contain the necessary data elements?

Are all necessary computations and processing steps performed the way you want them performed?

Are the types of ad hoc inquiries (the ones you would like to make) possible?

b. Capacity

Find out if the product will be able to handle your requirements for the size of files, number of data elements, number of table entries, volume of transactions, volume of reports and number of occurrences of certain data elements.

A CONTRACTOR

Determine whether response or turnaround time will be adequate.

Check all limitations.

c. Flexibility

Can transaction and report formats be changed easily?

Can screen layouts be changed easily?

How easy is it to add new computations or processing steps?

Can the programs be adapted to new applications?

d. Usability

Does the level of technical knowledge required to use this product properly match the level of knowledge of those who will be using it?

Is the user documentation complete, easy to read, easy to understand and up-to-date?

How readable, informative and easy to interpret are the reports and screen displays produced?

Is training available from the vendor?

e. Reliability

Does the product have a clear, modular design?

Has it been in actual use long enough to make sure that most of its bugs have been cleaned up?

How much of the system will become unusable when a part of it fails?

Does the product rely on any failure-prone hardware or noisy communication links?

Does the product incorporate any features for the detection and self-correction of errors?

Are there errors a user can make which will bring down the system?

What are the recovery capabilities?

f. Security

Does the product incorporate standard types of controls?

Does the product permit adequate back-up procedures?

Does it assist in any way to prevent intruders from extracting sensitive data from files or transaction input streams?

Does it help prevent employees from entering unauthorized transactions or running programs without authorization?

Does the product provide adequate detection and diagnosis of data entry and other types of errors?

Is a standard procedure provided that can be used to verify that the system is functioning properly?

g. Performance

At what rate will the product typically use machine cycles?

At what rate will it be requesting disk accesses? How much main memory will it require?

If it will be run on a virtual memory system, how much paging activity will result?

How many users can be on the system before it begins to bog down?

h. Serviceability

Are the source programs available? (Often they aren't).

If the vendor will be doing maintenance, how reliable and accessible is the company?

What level and quality of maintenance will the vendor supply? Is this guaranteed in writing?

Will changes to the system invalidate the warranty? Can the product easily be used in another operating environment? That is, is it portable?

If your staff will be doing the maintenance, are the programs written in a language with which your staff is familiar? Does it use techniques with which staffers may not be familiar?

Are sets of test data available with adequate documentation of how to use them and of what results to expect?

i. Ownership

What kind of rights to the product are you buying? Can you resell or rent the product to someone else?

If the vendor is making the product especially for you, will it also be marketed to others? If so, will the vendor pay you a royalty on each sale?

Are there restrictions on copying the product or its documentation?

Are there restrictions on the purposes for which the product may be used?

Will you be able to obtain full ownership rights and copies of the source programs if the vendor goes out of business?

CONTRACTOR OF

j. Minimizing Operating and Maintenance Costs

How much does the vendor charge for maintaining and upgrading the product?

How frequently will maintenance probably be required?

What is the cost per transaction of using this product?

What will the storage costs be?

Will purchase of this product require acquisition of additional hardware or personnel? How much will they cost on a continuing basis?

What will be the usable life span of this product?

k. Minimizing Purchase and Installation Costs

What initial costs are there besides the basic costs of the product?

Will you have to pay shipping costs? Will you pay transportation and lodging costs of the vendor personnel who will install the system and train your staff?

Will there be considerable delay between placement of the order and actual delivery of the product in a ready-touse state? How much will this cost?

What will conversion costs be?

<u>Any list</u> of objectives should include items from each of the ll categories. If any one of them is neglected, the adequacy of the evaluation must be suspect.

Reliability and programming costs are currently the <u>most</u> important criteria in software development for microprocessors. The primary objective is to develop a program

that works with a reasonable expenditure of time and money. The major cost in most  $\mu P$  projects is programming time; methods that can minimize the time required to complete a program are especially important. With the availability of larger memories, cheaper hardware, and faster processors, time, memory, and hardware constraints are not as critical in  $\mu P$  software development as they have been in the development of software for larger computers.

3. Ground Rules for Graphics Software Package Evaluation

The main reason to present separately this category of software is the tremendous importance of computer/graphics for Tactical Applications, especially in the area of Command-Control and Communications ( $C^3$ ). An example of their usefulness today is the use of image processing systems from Military Intelligence personnel, who use Image Processing to see through camouflage and detect objects of military interest in aerial and satellite photographs [22].

A typical computer/graphics system includes a computer (mini or micro), I/O devices and peripherals. The heart of this system is the software package that accepts graphics input and interaction and produces graphics output [21]. Therefore, in order to evaluate this software package, the following important issues must be carefully examined:

a. Simplicity

Features that are too complex for the application programmer to understand, indicate that there is no simplicity.

Anything difficult to explain in the user's manual will almost certainly be difficult to use.

b. Consistency

A consistent graphics system is one that behaves in a generally predictable manner (i.e., function names, calling sequences, coordinate systems, etc. should, without exceptions, follow simple and consistent patterns.

c. Completeness

The system must provide a set of functions that can conveniently handle a wide range of applications; missing functions will have to be supplied by the application programmer, because completeness does not imply comprehensiveness.

d. Robustness

The system, in the case of a serious error of the programmer, should report the error in the most helpful manner possible. Only in extreme circumstances should errors cause termination of execution, since this will generally cause the user to lose valuable results.

e. Performance

Graphics-system performance is often limited by such factors as O.S. response and display characteristics. The system has to maintain consistent performance, so that application programs provide an equally consistent speed of response.

f. Economy

Graphics systems should be small and economical so that adding graphics to an existing application program can always be considered.

#### 4. Expandability

For any microcomputer system, expandability is necessary. The minimum of 32 Kilobytes of RAM is required for the microprocessor to be considered at all. It has to be expandable to a minimum of 64 Kilobytes for serious consideration to be given for Navy use. This is based on projections of future applications requiring more memory in addition to immediate applications.

In evaluating the complete microcomputer system, it is important to examine the ability of the computer to support peripheral devices. Also, in order for the computer to be compatible with different terminals, printers, etc., the number of bits/sec (BAUD) must be able to be switched. The switching system, wires or toggles, must be able to be plugged into different positions on the board.

The computer has to have a serial and parallel port capability for support of both low and high speed peripheral hardware. It is desirable that the system have a mass storage unit capable of holding 1-15 megabytes of data, a terminal, and a printer.

It is also important, as it affects ease of maintenance, the way that the "mother board" can have additional boards added to it. For adverse environments, such as shipboard or airborne use, special springloaded connectors are necessary to insure reliable connections between removable units.

#### 5. Peripheral Devices

Among peripheral devices are included the I/O devices that will interface with the microcomputer (terminals, mass storage devices and printers).

When examining these devices, the following must be considered:

- (1) Number of bits per second that are transferred (baud rate).
- (2) Access speed of the device.
- (3) Amount of overhead necessary in the O.S. to support the device.
- (4) If the device, for an effective interface with the computer, needs serial or parallel ports.
  - a. Analysis of I/O Devices

(1) <u>Terminals</u>. There are two basic categories of terminals. The first of these, called "SMART," which comes with a built-in microprocessor and allows many applications without having to access the main computer (i.e., text processing). The second operates like a typewriter, for I/O between operator and computer.

Both categories come as "hard copy" terminals (including built-in printer) or video terminal (without printer).

Special characteristics to be examined:

- (a) Send and receive lower and upper case letters.
- (b) Include full 128 ASCII character set.
- (c) Video terminals require a minimum of 80 characters per line and 20 lines on the screen.
- (d) Hard-copy terminals are required to have capability of 132 characters per line.

It is underlined here that the "video" terminal is faster than the "hard copy" terminal.

(2) <u>Mass Storage Devices</u>. In this category belongs: hard-disk units, "floppy" disks, cassettes, and paper tapes. These units increase the capability of microcomputers by providing a virtual memory of up to several million bytes.

Special characteristics to be examined:

- (a) Memory capacity
- (b) Access speed
- (c) Error rate
- (d) Pilferability potential
- (e) Method of computer interface
- (f) Required "baud rate"
- (g) Power requirements
- (h) Potential for information loss, due to unit failure caused by power fluctuations or rough seas.

(3) <u>Printer</u>. The type of printers that are available to interface with the microcomputers are: line, dot matrix, friction feed, band, daisy wheel, and teletypes. Special characteristics to be examined:

- (a) Number of characters per inch
- (b) Line speed
- (c) Print quality
- (d) Type of ports required for interfacing
- (e) "Baud rate" required.
  - 6. Operational Capabilities

It is desirable for the microcomputer system to have the following capabilities: a. To support a minimum of three users on line at any time (centralized data base).

b. To have a form of write protection, to prevent the data base from any inadvertent change.

c. Direct Memory Access (DMA), which allows a high speed device to communicate directly with on-line memory, is a desirable feature.

d. It is also desirable to have multiply or divide capability, or to have floating point arithmetic capability.

e. I/O interrupts, to control I/O operations, and supervisor call interrupts, which are necessary to perform actual I/O.

7. Throughput

Any meaningful evaluation of computer performance must be based on the execution times of typical programs in typical applications. For example, for the Z-8000, these applications may involve high-level language compilers, operating systems and large data-base management.

8. Life Cycle Cost

a. Definition

Life cycle cost is the methodology for combining the results of all other evaluations into a single evaluation criterion which would be realistic and meaningful to the decision maker.

Unfortunately, the concept of computer productivity or performance is confounded by the great flexibility, both in design and in use, of these machines. Computer characteristics or capabilities which are essential measures of performance for one user are often irrelevant to another. Additionally, recent architectural advances which allow software to perform previous hardware functions, and vice versa, further blur the concept of "Machine" performance [6]. A total cost model is shown in figure 15.

b. Life-Cost Levels

As it is shown in figure 15, life-cost is divided into five levels and each one of these into two or more phases [23].

At the first level, life-cycle cost is obtained by summing costs incurred during design, development, verification and validation, and the operation and maintenance.

The development phase cost involves documentation, test and implementation of hardware and software, level 2. Hardware implementation cost consists of the cost of acquisition, manufacture, and installation of microcomputer and interconnecting link(s), in case of multi-microcomputer systems.

Cost of data processing is the sum of items shown on level four, which includes the cost of peripherals and facilities. The interconnecting network cost breaks down into cost of links and persible traffic switches (depending on architecture) that may reside either at computer locations or intermediate facilities for relaying and/or switching.

Software cost is generally much more difficult to assess than hardware, and besides that if cost models exist,



Life-cycle cost associated levels and elements at each level for a microcomputer system. (From: "Distributed Micro/Mini Computer Systems," microcomputer system. Cay Weitzman, 1980). Figure 15

they should be used with caution, as this phase may cost the user 70% of the total cost.

c. Calculations

All costs associated with the four phases on level one can be expressed as follows:

$$C_{MCS} = \sum_{i=1}^{4} (C_{HW} + C_{SW})_i$$
(1)

where,  $C_{MCS}$  = total life-cost of microcomputer system  $C_{HW}$  = hardware cost associated with the ith life-cycle phase  $C_{SW}$  = software cost associated with the ith life-cycle phase.

The annual cost of the operational and maintenance phase can be expressed as a fraction of the sum of the total costs of the first three phases:

$$C_{MCS} = (1 + fY) \sum_{i=1}^{3} (C_{HW} + C_{SW})_i$$
 (2)

- where, f = annual operational and maintenance cost (expressed in "per-year" terms)
  - Y = microcomputer system life-cycle time period of operational service, number of years.

The hardware cost can be decomposed as follows:

$$C_{HW_{i}} = (C_{DP} + C_{FAC} = C_{NET})_{i}$$
(3)

where,  $C_{DP}$  = data processing hardware cost associated with phase i.

- $C_{FAC}$  = facilities cost associated with phase i.
- C<sub>NET</sub> = communications/interconnect cost associated with phase i.

The total computer hardware cost associated with the implementation aspect of the development phase, in case of multi-microcomputer systems, can be expressed as follows:

$$C_{DP_{2}} = \sum_{j=1}^{G} C_{DP_{2,j}}$$
(4)

where, G = total number of microcomputers used in the system.

 $C_{DP_{2,j}}$  = the total hardware computer cost.

Software cost can be decomposed into the cost of developing real-time (RT) and non-real-time (NRT) code, i.e., the software cost term of equation (1) becomes:

 $C_{SW_{i}} = C_{NRT_{i}} + C_{RT_{i}}$ (5)

where,

- C<sub>NRT</sub> = cost of non real-time machine language instructions (NRT MLI's) during phase i (note that the equation is equally valid using HLL for programming)
- C<sub>RT</sub> = cost of real-time machine language instructions (RT MLI's) during phase i.

The cost of developing NRT MLI's can be expressed by the following equation:

$$C_{\rm NRT} = C_{\rm NRT} I_{\rm NRT} \tag{6}$$

where,  $c_{NRT}$  = cost per non real-time machine language instruction

I<sub>NRT</sub> = number of NRT MLI's.

and similarly,

 $C_{RT} = c_{RT} I_{RT}$ (7)

where,  $c_{pT} = cost per RT MLI$ 

 $I_{PT}$  = number of RT MLI's.

The total software required during the development and life cycle of the system can be divided into three major categories:

(1) <u>Application</u>, which is the operational software that supports user requirements (i.e., sensors, time table, etc.).

(2) <u>Test</u>, which serves to test the applications software at various levels of development (from algorithm to process) and during integration and validation testing (system level testing). Test software may consist of simulation software used to exercise and test the applications during the development phase by simulating the data processing subsystem, environment, and sensor subsystem and of data reduction and analysis of similar test tool programs, depending on the particular system application.

(3) <u>Support</u>, which consists of programs specially developed in support of a particular system including programs for configuration management, system maintenance, etc.

Hence, all the software required for a microcomputer system can be expressed as follows:

$$SW = SW_{APPLIC} + SW_{TEST} + SW_{SUPPORT}$$
(8)

Similar to previous groupings, each element can be further broken into real-time and non real-time software.

The cost per machine language instruction (MLI) has been found, from practical experience, in 1977 to be typically as follows [23].

$$c_{\rm NRT} = \frac{\$42}{\rm NRT \ MLI}$$

$$c_{\rm RT} = \frac{\$120}{\rm RT \ MLI}$$
(9)

The above rates include the following charges:

- ° Direct labor.
- ° Overhead.
- ° General and administrative.
- ° Fee.

والمراجع المراجع والمحافظ والمراجع والمحافظ والمحافظ والمحافظ والمراكب والمحافظ والمراجع والمحافظ والمراكب

- ° Computer time.
- \* Documentation.
- ° Travel.

The rates apply to the following major development phases:

- \* Requirements definition and analysis.
- ° Preliminary design.
- ° Code and debug.
- \* Testing (algorithm, routine, task subprogram, program, system integration, evaluation and validation, and acceptance by user).

d. Physical Dispersability and Survivability

Physical dispersability and survivability are closely related. Survivability is defined as the probability that a multi-microcomputer or minicomputer system can sustain hardware losses due to hardware failures, software errors, and hostile action and continue to carry out the nominal mission objective, as spelled out by the performance requirements, without degradation. Survivability is thus related to the probability that a system will fail during a particular time interval based on a certain system availability [23]. This attribute is important for systems that may be subject to destruction (i.e., vandalism, sabotage, or other forms of hostile action such as in military environments).

Survivability can be achieved using redundant, dispersed computers and transmission links between computers that are less susceptible to damaging effects or events, e.g., by operating in ways that reduce the probability of damage and by using redundancy techniques to increase the probability of maintaining the required connectivity of the system.

Figure 16 summarizes the relative importance of the key design attributes in some typical multi-microcomputer and minicomputer application areas.

e. Modularity

Modularity for growth is synonymous with adaptability, enhancement, extensibility, changeability, and modifiability. Growth can often be equated to design specifications and, in particular, design specifications for systems where

· ·

Application anter Attributes	No.	A REAL	To The	the second			Bull screenidion			./
Performance (throughput, response, etc.)	м	н	•	M	H/M	n/m	<b>H/</b> M	н	н	
Anoilability	M		н	M	L	#	*	н	м	
Reliability	м	н	M	M	L	н	6	н	2	
Full tolerance falls, switching elemands, water, alc.)	L	н	н	н	L	н	м	н	L	
Life cycle cost (off the shell hardware, software, ease at developments, etc.)	н	м	н	н	н	н	н	м	м	
Modularity/growth	L	м	м	н	н	н	н	н	н	
Form factor foulume, weight, power)	н	н	L	L	L	м	L	н	L	
Physical dispersability and survivability (failure acconfigureability)	L	н.	м	м	L	M	M	н	L	

Fig.16 Examples of potential application areas and the relative weighting of attributes for a distributed microor minicomputer system(H=very important, M=moderately, L=unimportant or incosequential).



Figure 17. Comparison of two potential distributed computer systems, A and B. As it is seen when A expanded to 3 times its original performance level, is 67% more costly than B. (Both fig.16,17 are from: "Distributed Micro/Minicomputer

systems", Cay Weitzman, 1980)

the exact future needs are difficult to predict [23]. Growth requirements are nevertheless imposed to minimize the system life-cycle cost. This is best illustrated by an example, as shown in figure 17.



#### V. CASE STUDY

The case study is presented to illustrate by means of a concrete example how the performance requirements are expressed in terms of parameters which are used to estimate the performance of any particular computer system.

The attack aircraft A6-E tactical system was chosen because it serves as a realistic representative system which exists and is suitably documented in references [30] and [31]. Also, the A6-E tactical system provides an example which requires the use of multi-microprocessor system instead of uniprocessor system.

This case study therefore serves to illustrate how the performance analysis is extended to multi-microcomputer systems.

#### A. THE A6-F OPERATIONAL FLIGHT PROGRAM

The operational flight program of the A6-E aircraft performs the following functions:

a. Navigational calculations.

b. Tracking and ranging calculations.

c. Ballistics calculations.

d. Sensor I/O and steering updates.

1. Description of the Functions

a. Navigational Calculations

The sensors used to generate information for the navigational system are:

- (1) Inertial navigational
- (2) Doppler radar
- (3) Magnetic compass
- (4) Airspeed indicator

(5) Altimeter.

The computer program is subdivided into nine segments, each of which is documented as a flowchart page. Table IVa contains the breakdown of each page of the flowchart with a count of each type of instruction, as well as the number of calls to library subroutines.

b. Tracking and Ranging

The purpose of this segment of the program is to establish the position of the target with respect to the aircraft. Tables IVd and IVe give the performance requirements for these calculations.

c. Ballistics Calculations

These determine, from the initial conditions at release, the down range travel and the time of fall of any particular weapon. The ballistic calculations are described in Table IVc. Table IVf contains the attack decision-making process.

d. Sensor 1/0 and Steering

This segment of the program controls the analog to digital converter. Correction signals to inertial navigation unit, radar antenna control, display control and steering commands are generated by this segment. Table IVb contains the performance parameters.

#### 2. Explanations for Tables IV (a-f)

The floating point operations were chosen to characterize the program, because of the programmer convenience for using floating point operations, whenever available. Because floating point hardware is becoming less expensive and more efficient, programmer convenience is considered more important than the additional hardware expenses.

The column headings refer to the instruction types categorized as follows:

° C - Conditional branch for integer operands

° L/S - Load or store an integer

- ° CF Conditional branch for floating point operands
- ° LFS Load or store a floating point operand
- \* FAD Floating point addition

• FMU - Floating point multiply

- FDV Floating point divide
- ° CO Cosine function
- ° SI Sine function
- AT Arc tangent function
- ° LN Logarithmic function
- ° SQ Square root function

The two rows of numbers, corresponding to each flowchart page, are the instruction counts in the most frequently executed execution path in the upper row and the most time consuming execution path in the lower row.

The two rows of totals at the bottom of the page accumulate the upper rows for the most frequent execution path, and the lower rows for the worst case execution path.

	<u>A6-E</u>	Nav	igat	iona	<u>1 Fu</u>	ncti	on Co	omp1	exi	<u>ty</u>	Meas	ures
	C	L/S	CF	LFS	FAD	FMU	FDV	ω	SI	AT	LN	sq
AIR DATA	1 3	9	3	30	10	7	2	0	0	0	1	0
	6	12	3	40	11	8	5	0	0	0	2	0
AIR DATA	2 1	7	0	23	10	7	2	2	0	0	1	1
	2	10	0	33	14	10	2	2	0	0	0	2
AIR MASS	1	3	6	29	11	4	4	1	1	2	0	1
ANGLES	4	6	6	38	11	4	4	1	1	2	0	1
DOPPLER	1	5	0	26	16	6	0	1	1	0	0	0
VELOCITY	2	6	3	33	17	6	1	1	1	0	0	0
SVCTEM	4	2	0	37	10	10	1	1	1	1	0	7
VELOCITY	6	6	0	62	16	10	2	2	2	1	0	3
RADO TN	2	2	7	22	0	6	2	٥	2	0	0	0
VERT LP	4	2 6	5	52	9 11	7	2 3	0	2	0	0	0
THEOTIAL	2	7	0	27	11	0	2	7	2	2	0	2
ANGLES	2	.3	0	27	11	8	2	1	2	2	0	2
	4	7	2	76	7.4	17	-		0	0	0	0
PLATFORM	1	2	2	30	14	15	5	1	0	U	U	U
UKKEUII	1	/	2	40	14	12	5	T	U	U	0	U
PLATFORM	1	1	0	58	27	22	6	1	1	0	0	0
CORRECT2	1	1	0	66	31	24	8	1	1	0	0	0
TOTALS	16	35	14	288	118	83	24	8	8	5	2	7
	28	57	19	391	136	94	32	9	9	5	2	8

TABLE IVa

Here and the second

TABL	E .	[V]	b
------	-----	-----	---

A6-E Input	Output and	l Steering
------------	------------	------------

	С	L/S	CF	LFS	FAD	FMU	FDV	CO	SI	AT	LN	SQ
COMMAND	3	11	0	41	7	3	2	3	1	0	0	0
STEERING 1	4	30	0	44	7	3	2	3	1	0	0	0
COMMAND	3	3	1	39	24	7	2	1	2	2	0	0
STEERING2	5	13	4	68	31	20	5	1	4	2	0	0
COMMAND	3	13	3	35	5	6	5	0	0	0	0	0
STEERING3	3	21	3	49	6	7	5	0	0	0	0	0
SAMPLE	0	52	1	0	0	0	0	0	0	0	0	0
INPUTS	0	56	1	0	· 0	0	0	0	0	0	0	0
INTERRUPT	4	20	0	4	2	0	0	0	0	0	0	0
SERVICE	9	32	0	16	7	0	0	0	0	0	0	0
STEERING	4	8	1	44	10	20	1	1	1	0	0	0
DISPLAY	5	10	1	48	12	21	1	2	1	0	0	0
DISCRETE	0	86	2	2	0	0	0	0	0	0	0	0
OUTPUTS	0	90	2	2	0	0	0	0	0	0	0	0
	•	• •	-		-		•					-
STEERING	2	10	0	6	1	1	0	0	0	0	0	0
KEY SEL1	5	31	3	22	1	1	0	0	0	0	0	0
STEEDINC	c	6	0	19	0	2	0	٥	Δ	0	0	٥
VEV CEL 2	J 10	10	1	10	0	2	0	0	0	0	0	0
REI SELZ	τu	19	T	00	U	3	U	U	U	U	U	U
TOTALS	24	209	8	189	49	39	10	5	4	2	0	0
	41	302	15	315	64	55	13	6	6	2	0	0

. Set and

1	AD-	AU91 09 Lassifi	95 NAT A J JUP	VAL POS METHOD N N 80 M	FOR EVAL	TE SCHO LUATION KAS	OL MON OF MIC	TEREY C Rocompu	A TERS FO	R TACTI	CAL APF	F/G C	9/2 ETC(U)	N.
		2.3 <b>)</b> 4.9 5 -												
														;
													END PATE FILMED 42-80 DTIC	
													لننا	



TABLE	IVc
-------	-----

Witches

and the second second

## A6-E Ballistics Function

	С	L/S	CF	LFS	FAD	FMU	FDV	ω	SI	AT	LN	SQ
ROCKET	0	0	4	48	17	19	5	1	1	0	0	0
ATTACK1	1	5	4	52	17	19	5	1	1	0	0	0
ROCKET	1	18	2	34	17	13	5	1	0	0	0	1
ATTACK2	1	19	2	36	17	13	5	1	0	0	0	1
BOMB	2	4	0	16	2	5	1	0	0	0	0	1
ATTACK1	3	5	0	30	5	9	1	0	0	0	0	1
BOMB	5	5	2	46	21	15	6	2	2	0	0	0
ATTACK2	5	5	4	90	23	17	6	2	2	0	0	0
BOMB	2	2	0	78	43	39	4	3	3	0	0	2
ATTACK3	2	2	1	93	43	40	4	3	3	0	0	2
BOMB	2	4	4	48	23	13	2	1	1	0	0	1
ATTACK4	3	5	6	58	23	13	2	1	1	0	0	1
BOMB	2	2	3	17	3	2	1	1	0	0	0	0
ATTACK5	3	4	6	36	3	3	2	1	0	0	0	0
BOMB	4	10	6	8	3	2	1	0	0	0	0	0
ATTACK6	6	14	9	27	6	6	1	0	0	0	0	0
BOMB	4	4	3	33	11	12	4	1	0	0	0	0
ATTACK7	11	11	4	76	18	14	5	2	0	0	0	0
COMMON	2	2	۵	84	41	42	3	0	1	0	0	1
DRAG	4	6	1	85	41	42	3	0	1	0	0	1
TOTALS	24	51	24	412	181	162	32	10	8	0	0	6
	39	76	37	583	196	176	34	11	8	0	0	6

		<u>A</u>	<u>6-E</u>	Trac	cking	and	Ran	gin	g Fu	inct	ion	
	С	L/S	CF	LFS	FAD	FMU	FDV	ω	SI	AT	LN	SQ
GREAT CIRC	0	0	0	46	10	13	4	4	5	2	0	0
NAVIGATION	0	0	0	46	10	13	4	4	5	2	0	0
TRACK RADR	3	7	5	23	6	6	0	0	0	0	0	0
TESTS	4	10	5	27	6	6	6	0	0	0	0	0
DEPR ANGLE	3	4	7	21	4	3	1	0	1	0	0	0
TRACK-1	3	16	7	21	4	3	1	0	1	0	0	0
TRACK SCAN	7	13	0	44	9	14	2	4	6	2	0	0
TESTS	9	21	0	51	10	14	2	4	6	2	0	0
DEPR ANGLE	5	9	0	11	2	2	2	0	0	0	0	0
TRACK-2	9	21	0	32	10	7	4	0	0	0.	. 0	0
LINE OF	2	8	4	31	9	5	4	3	1	1	0	0
SIGHT RNG1	3	12	5	50	10	6	5	4	1	1	0	0
LINE OF	10	25	2	16	2	2	0	1	1	0	0	0
SITE RNG2	10	45	3	19	2	2	0	1	1	0	0	0
RADAR	7	11	0	· 18	7	3	3	2	1	0	0	0
RANGING	9	15	0	18	7	3	3	2	1	0	0	0
SHRIKE	4	14	4	36	11	9	3	1	1	0	0	0
RANGING	15	39	0	25	9	3	3	3	3	1	0	1
TOTALS	41	91	22	246	60	57	19	16	16	5	0	0
	6 <b>2.</b>	179	25	289	68	57	28	17	18	6	0	1

### TABLE IV d

76

....

97

### TABLE IV.e

## A6-E Target Updates

	С	L/S	CF	LFS	FAD	FMU	FDV	CO	SI	AT	LN	SQ
TARGET	6	б	0	120	4	0	0	0	0	0	0	0
INI	7	38	0	140	6	1	0	0	1	0	0	0
TARGET	5	10	0	57	24	14	8	0	0	0	0	0
FILTERS1	7	14	0	73	28	18	11	0	0	0	0	0
	_					•	-			-	_	
TARGET	2	4	0	16	4	0	0	0	0	0	0	0
FILTERS2	2	4	0	16	4	0	0	0	0	0	0	0
SIEW	5	24	n	2	٥	2	0	n	0	0	0	0
IDDATE!	5	24		20	0	2	0	0	0	0	0	n n
OFDAILI	3	20	U	20	U	2	U	U	U	U	U	U
SLEW	8	12	2	49	16	10	4	1	1	0	0	0
UPDATE2	14	24	2	87	18	10	4	1	1	0	0	0
ANGLE	2	4	4	44	12	13	5	2	1	0	0	0
RATES	2	7	5	62	13	16	6	2	1	0	0	0
CURSOR	2	2	1	87	49	32	12	4	4	3	0	2
UPDATES	6	8	1	125	52	37	14	5	4	4	0	4
	•	10	-	24	•	•	•	•	•	•	•	•
KALIAK	1	10	ა 7	20	U	0	0	0	0	0	U	0
OUIPUIS	1	10	3	20	0	U	U	U	U	U	U	U
TARGET	1	3	0	46	15	11	3	1	1	2	0	1
POS	1	3	0	46	15	11	3	1	1	2	0	1
UPDATES	-	•	~				•	-	-	-	•	-
TOTALS	32	75	10	453	124	82	32	8	7	5	0	3
	45	134	11	595	136	95	38	9	8	6	0	5

### TABLE IV.f

## A6-E Attack Decisions

	С	L/S	CF	LFS	FAD	FMU	FDV	00	SI	AT	LN	SQ
RESELECT	8	9	0	0	0	0	0	0	0	0	0	0
LOGIC1	9	31	0	0	0	0	0	0	0	0	0	0
RESELECT	5	11	0	0	0	0	0	0	0	0	0	0
LOGIC2	14	30	0	4	0	0	0	0	0	0	0	0
RESELECT	7	8	0	0	0	0	0	0	0	0	0	0
LOGIC3	11	25	0	3	0	0	0	0	0	0	0	0
RESELECT	10	14	0	0	0	0	0	0	0	0	0	0
LOGIC4	20	36	0	0	0	0	0	0	0	0	0	0
RESELECT	8	10	0	0	0	0	0	0	0	0	0	0
LOGIC5	16	30	0	0	0	0	0	0	0	0	0	0
ATTACK	7	11	3	15	2	2	0	0	0	0	1	0
SELECT1	8	25	3	25	6	3	1	0	0	0	1	0
ATTACK	6	23	0	22	5	5	1	1	1	0	0	0
SELECT2	10	40	0	38	7	5	1	1	1	0	0	0
STEP OUT	2	2	7	15	6	1	1	0	0	1	0	0
OF ATTK	9	19	10	18	6	1	1	0	0	1	0	0
ATTACK	5	7	10	16	4	2	0	0	0	0	0	0
VALID1	6	34	11	17	5	2	0	0	0	0	0	0
ATTACK	7	9	4	24	12	2	2	0	0	0	0	0
VALID2	10	20	5	39	18	3	2	0	0	0	0	0
TOTALS	65	104	24	92	29	12	4	1	1	1	1	0
	113	290	29	144	42	14	5	1	1	1	1	0

99

B. 23

Tables IV(g-1) tabulate the number of FORTRAN or CMS-2 instructions (if the flowchart were translated into FORTRAN or CMS-2) in the categories of arithmetic (AR), conditional (IF), and control alteration (GO) statements. In addition, the number of assembly language instructions in the actual program and the number of bytes (8 bits) the program occupies in memory are also tabulated.

The entire functional requirements of the A6-E operational flight program are expressed by table IVm.

The purpose of all these tables IV(a-m) is to help the reader to find the components which are used in the calculations of this study for further use.

	<u>A6-E</u>	Navigatio	n	Higher	level	Language	Complexity
		AR	ĨF	GO	TOTAL	ASSEMB	LY BYTES
AIR DATA	1	19	9	6	34	118	242
AIR DATA	2	20	2	1	23	87	168
AIR MASS ANGLES	•	15	10	2	27	124	272
DOPPLER VELOCITY	•	16	5	4	25	90	176
SYSTEM VELOCITY	•	26	4	4	34	115	232
BARO INF VERT LOO	RT PP	25	9	6	40	127	270
INERTIAL ANGLES	4	16	3	2	21	78	168
PLATFORM CORRECTI	t Ions I	17 1	2	2	21	100	200
PLATFORM CORRECTS	1	28	1	1	30	243	488
TOTALS		182	45	28	255	1082	2216

TABLE	IVg
-------	-----

C

### TABLE IV.h

# A6-E Input/Output and Steering Higher Level Language Complexity

	AR	IF	GO	TOTAL	ASSEMBLY	BYTES
COMMAND STEERING 1	33	4	4	41	109	242
COMMAND STEERING 2	33	9	7	49	188	430
COMMAND STEERING 3	28	6	5	39	99	318
SAMPLE INPUTS	52	1	1	54	272	482
INTERRUPT Service	30	8	7	45	147	376
STEERING DISPLAY	23	6	4	33	127	252
DISCRETE OUTPUTS	46	2	2	50	254	552
STEERING KEY SEL 1	18	8	5	31		
STEERING KEY SEL 2	38	10	10	58	255	588
TOTALS	301	54	45	400	1451	3240

102

*.*: .

2.3

<u>A6-E</u>	Ballistics	Function	Higher	Level	Language	Complexity
	AR	IF	GO	TOTAL	ASSEMBLY	BYTES
ROCKET ATTACK	1 26	5	3	34	113	252
ROCKET ATTACK	2 16	2	3	21	100	216
BOMB ATTACK	1 13	3	3	19	62	136
BOMB ATTACK	2 38	9	2	49	240	480
BOMB ATTACK	3 29	3	3	35	261	532
BOMB ATTACK	4 22	9	4	35	164	356
BOMB ATTACK	5 15	9	7	31	73	168
BOMB ATTACK	6 17	14	7	38	98	224
BOMB ATTACK	7 38	15	11	64	229	498
COMMON DRAG	21	5	3	29	209	434
TOTALS	235	74	46	355	1549	3296

TABLE IV.i

.

P. 4 8.5
# TABLE IV.j

	Higher	Level	Language	Complex	city	
	AR	IF	GO	TOTAL	ASSEMBLY	BYTES
GREAT CIRC NAVIGATION	14	0	0	14	82	164
TRACK RADAR TESTS	13	9	7	29	78	174
DEPR ANGLE Tracking 1	10	12	10	32	102	250
TRACK SCAN TESTS	21	9	5	35	157	338
DEPR ANGLE TRACKING 2	22	9	9	40	112	268
LINE OF SIGHT RANGE 1	26	8	4	38	159	322
LINE OF SIGHT RANGE 2	19	13	7	39	107	244
SHRIKE RANKING	28	13	; 8	49	138	300
RADAR RANGING	23	13	5 10	46	131	284
TOTALS	176	5 86	5 60	322	1066	2344

A6-E Tracking and Ranging Function Higher Level Language Complexity

104

|--|

Ab-E Target	Upda	tes H1	gner	Level La	nguage Comp.	lexity
	AR	IF	GO	TOTAL	ASSEMBLY	BYTES
TARGET INITIALIZE	50	6	5	61	152	306
TARGET POS FILTERS 1	43	7	5	55	216	490
TARGET POS FILTERS 2	10	2	1	13	15	36
SLEW Update 1	17	5	5	27	50	102
SLEW UPDATE 2	46	16	8	70	189	396
ANGLE RATES	27	7	3	37	136	284
CURSOR UPDATES	38	7	3	48	316	678
RADAR OUTPUTS	15	4	3	22	86	230
TARGET POS UPDATES	18	1	0	19	88	176
TOTALS	264	55	33	352	1248	2698

,

105

- 21

TABLE	IV	.1
-------	----	----

	A6-E Attack	Decis	ions	Higher	Level	Language	Comple	xity
		AR	IF	GO	TOTA	L ASSEI	<b>(BLY</b>	BYTES
RES LOG	SELECT	14	10	5	29	7		
RES LOG	SELECT SIC 2	24	15	12	51	The these	totals e five	for lines
RES LOG	SELECT SIC 3	21	11	6	38	appea	ar belo	)W
RES LOG	ELECT IC 4	14	20	8	42			
RES LOG	ELECT IC 5	10	16	6	32	ر 5	83	1324
ATT SEL	ACK ECT 1	29	11	4	44	10	51	390
ATI Sel	ACK ECT 2	28	10	6	44	10	00	236
STE OF	EP OUT ATTACK	11	19	5	35		81	208
ATI VAL	ACK ID 1	25	17	12	54	14	45	342
ATT VAL	ACK ID 2	26	15	5	46	14	18	360
тот	ALS	202	144	69	415	12	L 8	2860

106

Service and the service of the servi

TABLE	IV	<b>.</b> m
-------	----	------------

	I	cstru	ction	s	Vari	ables	Ext	ernal
Subprograms	S	M	L	Х	Int	Real	Int	Real
Navigational	51	302	225	29	18	125	3	47
Function	85	405	262	31	18	125	3	47
Tracking ६	239	730	374	64	50	192	7	7
Ranging	433	954	439	71	50	192	7	7
Ballistics	234	552	420	28	20	170	18	16
Calculations	518	793	467	29	20	170	18	16
Sensor I/O &	233	198	98	11	87	103	17	16
Steering	343	330	132	14	87	103	17	16

### Summary of A6-E Program Segments

The headings of this table are described as follows:

- ° S Short instructions, 16 bit integers.
- ° M Medium length instructions: load, store, and compare floating point quantities.
- ° L Long floating point instructions: FAD, FMU, FDV.
- X Subprograms which calculate sines, cosines, etc.
- ° INT Number of integer variables in the program.
- ° REAL Number of floating point variables in the program.
- EXT Number of variables which are used by other programs external to the named one.

107

and the second

### B. PERFORMANCE ESTIMATES FOR UNIPROCESSOR SYSTEMS

The attached FORTRAN program was written in order to estimate the average and the worst case performance for each of six major functions of the A6-E operational program.

The performance requirements are expressed in terms of the fundamental operations listed as columns in the tables IV(a-f). The first row of each row pair corresponds to the number of operations required in the most frequently executed path of each subfunction. The second row of the pair corresponds to the path which requires the maximum or worst case execution time. The totals at the bottom of each page accumulate the corresponding values of each subfunction to give the totals for the major functions.

The performance estimate of any computer is calculated by multiplying the execution times of the basic operations for that particular computer, with the number of times that basic operation is executed. The worst case execution time is determined for one execution of the major function by taking the totals in the bottom row of each table and multiplying these values by the corresponding execution times of the specified computer.

Each of the major functions must be executed a specified number of times per second. The repetition rate depends on the function and varies from fifty times per second (50 HZ) to five times per second (5 HZ). In order that the performance of a computer be acceptable, the worst case execution times multiplied by the repetition rates must not exceed

108

6 7**3 8** 

REAL#4 NAME(12), RES(6,3),NM(3,12),CCMF(4),TOTRES(3) INTEGER TIME(6,12),ANSWER CCCC THE FOLLOWING DATA REFER TO TOTALS OF EACH INSTR. CATA TIPE/28.41.35.62.45.113.57.302.76.179.134.290. 115.15.37.25.11.25.351.315.583.285.595.144. 1136.64.156.68.136.42.94.55.176.57.55.14. 132.13.34.28.38.5.5.6.11.17.9.1. 19.6.8.18.8.1.5.2.0.6.0.1. 12.0.0.0.0.1.8.0.6.1.5.0/ COLUMN HEALINGS : C CONDITIONAL BEANCH FOR INTEGER OPERANOS L/S LGAD OR STORE AN INTIGER CF CONDITIONAL BRANCH FOR FLOATING POINT OPERAND LFS LCAD GR STORE A FLOATING POINT OPERAND FAC FLOATING POINT ACCITION FMU FLOATING FOINT MULTIPLY FDV FLOATING FOINT MULTIPLY FDV FLOATING FOINT CIVICE CO COSINE FUNCTION SI SINE FLACTION AT ARC TANGENT FLACTION N LCGARITHMIC FUNCTION SO SGUARE FOOT FUNCTION SCUARE FEET FUNCTION SO CATA NAME/" C "." L/S"," CF "," LFS"," FAD"," FML", 1 FDV", CC "." SI "." AT "," LN "," SC "/ ເ ເບັ້າ CENTINUE FORMAT(12A4) REAC(5.510)CCMP REAC(5.510)CCMP READ(5.201)((N\*(I.J),I=1.3),J=1.12) FGRMAT(3F10.2) FCRMAT(///,11X.\*INSTRUCTION\*,E),\*EXECUTION TIMES FCR\*, 1//.25X.\*AN/UNK-7\*,3X.\*Z-8000\*,3X.\*INTEL-8086\*,/, 111X.\* WRITE(6.22) 310 201 22 CALCULATION OF EXECUTION TIME FOR EACH OF THE FOLLOWING SIX(6) SUEFFOERAMS : NAVIGATIONAL I/O AND STEERING BALLISTICS TRACKING AND RANGING TARGET UPDATES ATTACK FOOTSIONS ATTACK DECISIONS CALCULATION OF EXCLISION TIME OF FACE SUPPROGRAM E0 401 J=1.3 TGTRES(J)=0.0 E0 100 J=1.6 E0 10C I=1.3 RES(J.I)=C.C E0 102 K=I.12 RES(J.I)=RES(J.I)+TIME(J.K)\*N\*(I.K) 401 102

-

Ç	CAL	.cu	L	A 1	Ī	c	N	С	F	T	<b>'C</b>	T,	۹٤		£	χł	EC	U	T I	: <b>C</b>	N	Ţ	Ī	NŚ	-										
C 100	)	ŢC	Ţ	R	S	(	I	) =	T	DT	R	É	s (	1	1	<b>+</b> F	24	S	( )	۱.	I	)													
555		L= DO WR FC	I I R	39	99 E(	6			• []•	12 )N 12	A X	۱ م د و	E (	ļI	) 3	• ^ X •	, M , F	( 1	1, D.	12	);	• N 2 (	2	( ; x ;	2 . , F	I	). 3.	1	M ( } )	3	•	I)			
399 Ç	}	CG	N	. I	. N	U	E:																												
111		WR	Ī	ŢĘ		é		ļi	i	)												•				~		-,	~ •		-				
1 20	์ 1		•		N	>	Ľ	ĸ	-	<u> </u>	•	52	X .	; 1	z	- 8	10 10	0	5	2 2	4)	Κ,	•	1	Ť	5		8	) e	6		•/	; ' ; ; 		
	i	- -		15	-	-			-		-		_	)			-						-											•	
95	1		R   2		F	1	]:	1 2	)	, 6 , )	•	F	F (	: 6	F	<b>A</b>		•	2×	<b>,</b>	12	2.	3	X	, •	T	¢K	2	s I	'•	3	Χ,	F1]	•2	•
200			N			ć		;	),	J .	• (	F :	55 • 1	5 ( . r	J	, l	. )	• <sup>i</sup>	L= v=	:1	•	3) ••	r		7	Ŧ			1			<b>c</b> 1 ·			
400	<b></b> 1	2(	12	T II	F		2	2	j	, , , ,	T	• r ·	- 1 1 C	: F	с I I	4 L	- i )	-	/~ .1=	יי 1:	ر ب	• 1 3 1	L	n	1	Ŧ	r C		* 1		•	-14	C • C	•	
444		FO			t (	۲ 6	,	j i 4	) 4	) )		•	• •		•	••	• •		<b>u</b> -		•.	.,													
		EN	C																																

\$ENTRY

## INSTRUCTION EXECUTION TIMES FOR

	AN/UYK-7	2-8000	1NT=L-8186
С	2.00	2.00	1.00
L/S	1.50	0.75	4.23
CF	3.00	10.00	10.00
LFS	3.00	10.00	13.00
FAD	£.25	7.50	\$.00
FMU	10.00	22.50	16.00
FDV	17.00	48.75	<b>39.</b> 35
CO	162.50	137.50	110.00
SI	162.50	137.50	116.00
AT	162.50	137.50	110.00
LN	200.00	162.50	130.00
<b>S</b> Q	15.00	45.10	26.))

RESULTS	IN M	ICRGSEC)	FCR: AN/LYK-7	Z-8000	INTEL 8086
PROGRAM		TAKES	7963.66	12741.25	12771.00
PROGRAM	2	TAKES	4971.14	7884.75	\$107.09
PROGRAM	3	TAKES	8792.50	16305.00	16656.CC
PROGRAM	4	TAKES	5483.CC	12238.25	12061.00
PROGRAM	5	TAKES	8367.50	14648.0)	15552.00
PROGRAM	6	TAKES	2355.((	3622.25	4720.00
TOTAL EX	ECUT	ION TIME	41932.00	67439 <b>.</b> 5C	71067.00

one second. If one second is exceeded, only a multicomputer solution is possible. The worst case execution time estimates for each major function are tabulated in table V. A uniprocessor performance evaluation is shown on table VI.

### TABLE V

### Execution Time Estimates for Each Major Function

Function	Rep. Rate		<u>M P U T E</u> 7-8000	R INTEL-8086
	(HZ)		Enhanced	Enhanced
Navigational	50	398150µs	637062 µs	638550μs
I/O and Steering	25	124275	197119	227675
Ballistics	Demand (about 30)	263775	489150	505680
Tracking and Ranging	12.5	118537	152978	150762
Target Updates	12.5	104594	183100	194400
Attack Decisions	5	11775	18111	23600
Total (in seconds)		1.021106	1.677520	1.740667

Notes: (1) Enhanced - includes a floating point chip (i.e., INTEL-8087).

(2) The above numbers represent the results of the execution times (coming from the attached FORTRAN program) multiplied by the repetition rate of each subprogram. The total execution time gives an indication if the system is acceptable (time less than 1 second) or not (in case of  $\mu$ C that means solution must come from multi-microcomputer system.) This case is presented in the following pages.

الالعادي

TABLE VI

!

Uni-Microprocessors Performance Evaluation

249	140	87	36	32	30	Price (in \$)
A 5V/300mA	5V/300m	SV/275mA	5V/250mA	12V/40mA	sv/90mA	Power requirements
8 64 pin DIP	40 or 4 pin DIP	40 pin DIP	40 pin DIP	40 pin DIP	40 pin DIP	Package
Q	en	£	3	None	S	Dedicated I/O control lines
1/11L	1/TTL	11/TTL	1/TTL	2/9 V	1.5 V	Clock phases/ voltage swing
1 MHZ	4 MHZ	8/5 MHZ	100 KHZ/2 MHZ	2/3 MHZ	5 KHZ/4.5 MHZ	Clock frequency
N/A <sup>2</sup>	(div) 90.0	(div) 37.8	(interrupt) 6	(swap) 9	(set bit at address) 5.75	Longest instruc- tion/time (in µs)
56	110+	67	59	78	158	Number of basic instructions
(many) 0.5	(many) .75	(many) .40	(many) 2	(add reg.) 2	(Read reg. to reg.) 1.0	Shortest instruc- tion/time (in µs)
2 to 10 bytes	2 to 6 bytes	l to 6 bytes	1 to 4 bytes	8 to 24 bits	1 to 3 bytes	Instruction word size
16 Mbytes	8 Mbytes/ 64 Kbytes	1 Mbytes	64 Kbytes	65 Kbytes	65 Kbytes	Addressing range
23 bits <sup>1</sup>	23/16 bits	20 bits	16 bits	16 bits	16 bits	Address bus size
16 bits	16 bits	16 bits	8/16 bits	8 bits	8 bits	Data word size
MC68000	28000	8086	MC6809	8080	2-80	SPECIFICATIONS

. . . . . .

.

C. PERFORMANCE ESTIMATES FOR MULTI-MICROCOMPUTER SYSTEMS

If the performance requirements exceed the processing capacity of the uniprocessor, then it is possible to implement the system by using multi-microcomputer systems. The most available architectures which are designed specifically for real-time applications are the single board computers





connected by a system's bus to common memory. Figure 18, above, depicts a typical architecture.

Because each Single Board Computer (SBC) contains its own memory for storing programs and private data, the system's bus is only used when external data is accessed. If the computers are dedicated to distinct functional tasks, each computer performs independently of the others and only external data needed for computation and synchronization of the processes is communicated on the system's bus. If the functional processes are relatively independent, the performance of the system is the sum of the performances of the component parts. If, however, common data is frequently accessed by all of the functional processes, the system's bus may become the bottleneck of such a multi-microcomputer system. In order to evaluate the system's bus performance, experiments were designed and the results are summarized below.

When the system's bus is used at 50% of capacity, no observable degradation in the system's performance is observed. At 75% of bus capacity, approximately 10% loss of the system's performance was observed. The system's degradation becomes serious above 80% of capacity.

Each application must be studied individually in order to partition the system into functional elements so that a single board computer is able to satisfactorily perform the computations and communicate information needed externally to common memory.

With the case study of A6-E operational flight program, the following partition, table VII of the functional program was found satisfactory.

The results of table VII are that using three (3) single board computers in parallel (INTEL 8086/8087) [33], the percentage of the system's bus use is very low (about 4.87% of

### TABLE VII

Functional	Partitioning	of the
A6-E Operat	ional Flight	Program

SBC	#		Functions	Estimated Execution Time/Sec. For INTEL-8086	External Variables Access Time	% of Bus Use
1			Navigational	0.64 sec	29100µs/sec	2.9%
2		0 0 0	I/O Steering Attack Decisions Target Update Tracking and Ranging	$\begin{array}{c} 0.23 \\ 0.02 \\ 0.19 \\ 0.15 \end{array}  0.59$	10731	1.07%
3			Ballistics	0.51	9000	0.9%
			TOTAL	1.74 sec		4.87%

Note: The value of External Variable Access Time includes the number of words per second multiplied by the repetition rate of the function and by the execution time for two more instructions of INTEL 8086, which is  $6.0 \ \mu s$ .

capacity) and therefore, according to the previous assumptions, this system is acceptable.

D. SUMMARY

The case study presents an implemented system, A6-E operational flight program, as a concrete example to illustrate the proposed method of computer evaluation. Whenever a system has been implemented, an accurate estimate of the computational requirements can be made. The estimates were based on the "worst case" computational requirements.

From table V, none of the uniprocessors have a computational capacity to match the requirements. A multi-computer solution, therefore, must be used. Table VII illustrates a three computer solution to the problem. Although a two computer solution would also have been possible for the INTEL 8086, the computational capacity of each computer would be used at 80% level. Future growth of the system would be severely limited. The three computer solution would allow less restricted future growth as well as more reliable performance. If any one computer fails, the two remaining computers can share the computational load without degradation in performance.

### VI. CONCLUSIONS

This thesis provided a method to evaluate microcomputer systems for Tactical Applications, presented a Case Study to illustrate by means of a concrete example how the performance requirements are expressed in terms of parameters which are used to estimate the performance of any particular computer system, and contained important information to aid in the evaluator's work, including tables in the Appendix which list microcomputer characteristics.

The purpose of this chapter is to emphasize the main difficulties, make recommendations, and draw conclusions about the use of microcomputer systems in tactical applications.

The evaluation of microcomputer performance capacities indicates that whereas the eight (8) bit microcomputers are deficient in arithmetic capability, the newest sixteen (16) bit microcomputers enhanced by floating point arithmetic units are close rivals to the Navy standard real-time computers, AN/UYK-7 and AN/UYK-20. Because the single board computers are designed to be building blocks of multi-microcomputer systems, they provide a greater flexibility for system's expansion and reliability. The computational capacity of multi-microcomputer systems on a system's bus can be easily expanded to meet the requirements of most presently conceived real-time combat applications.

The three key barriers to rapid introduction of microprocessors into Military and industrial systems relate to:

• Concerns for life-cycle supportability (for both military and industrial applications where anticipated system life is fifteen (15) years or longer) [26].

• The belief that the microcomputer performance in military applications is not adequate.

• The microcomputer technology is not compatible with previously defined architectural standards. Because of large prior investments in software tools and applications software geared to the established hardware standards, there is reluctance to undertake costly changes.

Certain recommendations can be made regarding the use of microcomputer systems in tactical applications. These are:

• Adaption of standard architectural families, enabling the development community to partially overcome [26] the development cycle delay and take advantage of the logistics maintenance and upgrade process, wherein old circuit boards are replaced with new boards containing microprocessors. The benefits in reliability, power, weight, cost, and selfdiagnosis capability could be significant.

• Initiating training programs designed to educate endusers of microcomputer systems. Microprocessors would certainly be used more widely in new designs if designers felt comfortable using them. Design aids, such as software development systems, logic analyzers, emulators, and High Order Languages will help. In addition, some stability in

languages (i.e., DOD instructions about Ada) and architectures is needed.

• Some microprocessor standardization can be beneficial and needs to be proposed, and the microprocessor industry is becoming increasingly willing to accept it. Central to this need for standards is user concern for lifecycle supportability of industrial and military electronic systems which contain embedded microprocessors and microcomputers. Standards, such as bus interfaces and protocols, language, component interchangeability, or functional description can play a key role in overcoming these concerns, as long as the standards are intelligently applied and not unilaterally decreed.

The major conclusions produced by this work are that single board microcomputer systems, and multi-microcomputer systems are a flexible and economical solution to tactical systems' implementations.

For those users who have no prior investment, the main criteria for selection of microcomputer-based systems are:

- Performance capacity of the processor.
- ° Software support from various sources.
- ° Manufacturer's risk.

For those users with prior investment, the following conversion costs need be considered:

- ° Change of High Order Language, or
- Compiler conversion.

The Armed Forces can influence the microprocessor/microcomputer industries in two ways:

- <sup>°</sup> By seeding money to influence basic technology directions, and
- \* By consolidating numerous small-quantity purchases into bulk buys, using functional standards to achieve commonality.

It is believed that the use of microcomputer systems, while considering the disadvantages which follow every new technology, will increase the effectiveness of weapon systems and will allow the Armed Forces to remain competitive or become superior to potential adversaries.

#### APPENDIX A

### REQUIRED TIME TO SELECT A MICROCOMPUTER

From previous experience, in the U.S. Navy, the time to select a computer system (other than microcomputer) fluctuates between 10 and 24 months [10].

Why does it take such a long time?

This is a reasonable question from many people without the necessary background in this field. Nevertheless, the answer to the above question is that the selection process must solve complex problems and, as the problems become more complex, so do the tools we employ. Complexity does not necessarily mean delay, but very often it accounts for a large part of the necessary time for the solution of the problem.

In the area of microcomputers, it is believed that the selection time is between 6 and 12 months, as outlined in Table VIII.

It is necessary for the selection of a microcomputer system, as for any other equivalent equipment, to form two individual groups: the Source Selection Advisory Council (SSAC) (consisting of the procurement contracting officer, the representative of CNO, and one or two members of the user), and the Source Selection Evaluation Board (SSEB) (consisting of the Project Leader and two or three technical advisors from the user, with necessary experience).

TABLE \	ľ	I	I	I
---------	---	---	---	---

### Required Time to Select a Microcomputer

	Selection Steps	Estimated Minimum	tim -	e in days Maximum
1.	Draft request for proposals for approved project	30	-	60
2.	Release of draft for comments	30	-	30
3.	Revision of request for proposals	20	-	20
4.	Response to request for proposals	40	-	80
5.	Evaluation of proposals and benchmarking	30	-	90
6.	Administrative time after evaluatio	on 20	-	45
7.	Installation of equipment after contract award	20	-	40
	TOTAL	180	- 3	65

The SSAC assures adherence to the headquarters policy and requirements. It reviews and approves the solicitation document, the selection plan, and the recommendations of the SSEB [10].

Responsibility for policy and specially configured hardware and programs for tactical systems belongs to the Tactical Digital Systems Office (TADSO) of CNO.

1. The Selection Process

The selection process has the following steps:

a. Request for Proposals (RFP)

As its name implies, this form invites firms to submit proposals to satisfy the requirements. This is a

formal document, with all the necessary descriptions of requirements, contractual terms and conditions, specific regulations, and finally a technical description of the requirements. The emphasis is on "requirements": data processing to be done, reliability, type of function wanted, etc.

b. Response to the RFP

The time for response from vendors depends on the complexity of the given requirements; usually this time is from 1 to 3 months. During this period the vendors are organizing "bid teams" and the user benchmarks. At the specified date and time, the RFPs are sent to the selection office. At this point it is underlined that corrections to the user's requirements means that an extension must be given to the vendors. It must also be kept in mind that even slight changes in wording may cause a protest or lead to an interpretation disadvantageous to the user.

c. Evaluation of Proposals

At this step, the following take place.

(1) From the responses received, which must arrive on time in order to be valid, is established how many vendors are going to bid.

(2) The user starts to work on the proposals.

State of the second

(3) If the proposal fails, after the first validation against technical literature, the vendor is declared nonresponsive and is notified that he will not be considered further (e.g., if the RFP requires the expansion of the main

memory and the vendor's technical justification proves that does not satisfy). By this way, the number of vendors for further evaluation is narrowed.

(4) The benchmarking, following the above step, gives an added measure for the evaluation.

d. Benchmarks

Benchmarking is only one of a large number of tools available to us for measuring the performance of machines and working out effectively which machine to get. Some tools for the assessment of computer performance are the following:

- Mathematical models, based on operations research or more specialized mathematical techniques (Markov model or decision theory).
- (2) Simulation models.
- (3) Programs designed to test specific functions.
- (4) Hardware monitors.
- (5) Software assessment, such as examining the operating system or using trace programs.
- (6) Cost analysis, which is kept entirely separate from the technical evaluation.

e. Selection

At this point, the winning vendor is selected, after the proposals have passed the technical evaluation, benchmarking, cost analysis, and check that both parties agree to the contractual terms and conditions.

f. Contract Award

This is the final step of the selection process. The user must verify the winning vendor that the business decision has been made, clearances and approvals have been obtained and the authorized person may award the contract.

The selection process is lengthy, costly, and difficult. However, it is necessary to follow step-by-step all the procedures described above if it is desired to avoid a failure.

#### APPENDIX B

### CHECKLIST FOR EVALUATING MICROCOMPUTER VENDOR POLICIES

### Payoff Idea

There are two sides to every coin. On one hand, selecting a microcomputer system for a specific application or job requires consideration of hardware and software performance. On the other hand, vendor support may be crucial in the form of services (both pre- and post-installation), hardware maintenance contracts, software maintenance practices, and educational programs. Buying a microcomputer based on hardware and software performance alone, without considering other vendor policies, is precarious. This paper presents pertinent categories for vendor comparison and a checklist for evaluating, any vendor's offerings. These questions can also be submitted to a vendor when a competitive analysis must be performed.

#### Evaluation Methodology

In order for buyers to evaluate vendor policies effectively, current information must be available. This paper assembles a checklist of questions that may be used to structure the information collection process. The buyer should first read the checklist in order to become familiar with the specific information used to compare vendor policies, and then choose those questions most pertinent for the desired project or application. Potential vendors should then be selected and asked to respond, in writing, to these questions, in sufficient time.

Using this checklist properly should guarantee thorough information collection, and define the scope and cost of services available [27].

### CHECKLIST QUESTIONS

The evaluator should request the following information from the vendor.

#### Hardware Sales

Policies regarding a manufacturer's pricing structure are one of the primary considerations made before purchasing any computer system. This section contains a series of questions pertaining to vendor sales posture, available discounts, trade-in allowances, Original Equipment Manufacturer (OEM) equipment resale, etc.

- 1. Do you rent, lease, or sell your equipment?
- 2. If you provide a lease, is it (a)
  - ° Monthly, within a 90-day cancellation?
  - ° One-year lease?
  - <sup>°</sup> Two-to-four-year lease?
  - Full payout lease?
  - Lease/other terms?
- 3. Do you offer quantity discounts to end-users on your equipment?
- 4. What is the range of your discounts for the following quantities:

#### End-user %

0	3 or less	
0	4 - 10	
0	11 - 25	
٥	26 - 50	
0	51 - 99	
٥	100 plus	

5. Where is the discount based? Specify.

6. Do you charge separately for:

- ° Hardware?
- Maintenance?
- Installation?
- ° Documentation?
- Program support?
- ° Education?

7. Do you pay for the shipping costs?

7a. What are your specified delivery times

- Immediately?
- ° 30 days?
- ° 60 days or less?
- ° 90 days or more? Specify.

8. Do you accept trade-in of old equipment on new?

8a. If yes, what is your allowance structure? Specify.

8b. Will you sell used and/or reconditioned hardware?

8c. If yes, how is it discounted? Specify.

9. Will you sell OEM designated equipment to an end user?

9a. Is the OEM equipment normally sold under your name or that of the OEM? Specify.

### Hardware Warranty

Another important consideration is the extent and breadth of the hardware warranty. The duration, the types of service included, etc., are important and useful.

1. Do you warrant your hardware?

1a. If yes, parts only, or parts and labor? Specify.

1b. If yes, for what length of time?

°30 days?

°60 days?

°Six months?

"One year or more? Specify.

2. When does the warranty begin?

<sup>o</sup>Date of shipment?

\*Date of receipt?

°Other? Specify.

- 3. Do you provide warranty service outside of normal working hours?
- 3a. If yes, is it billable or non-billable?

3b. Do you bill extra for travel?

- 4. Do you charge for situations in which no defect is found?
- 5. How is attached OEM equipment warranted? Specify.

### Hardware Maintenance

After the warranty period expires, contract and/or per-call maintenance must be carefully considered in order to plan for most contingencies. The availability of maintenance on foreign peripherals is also important, as are the type and cost of maintenance education that is offered.

°Nights?

Cases in which no trouble is found?
Operator errors?
Other? Specify.

2. Do you charge to correct a design defect?

3. Do you offer maintenance education?

3a. If yes, is it:

"Billable and/or non-billable?

<sup>°</sup>Available at the customer's location?

"Available at multiple locations? At how many?

#### Software Products

Software support is also a primary consideration, especially in terms of the cost of license fees and the extent of the license.

- 1. Is your firm's software copyrighted?
- 2. Do you sell or license your software?
- 3. Is there a license fee?
- 3a. If yes, how is it paid?

°Monthly? °Yearly? °One-time charge? °Other? Specify.

- 4. Do you offer software discounts if a customer upgrades from one licensed product to another?
- 5. Is software under development ever made available to users?5a. If yes, under what circumstances?

#### Software Distribution

The form in which the software is distributed is often important to the system user.

- On what media does your firm distribute software?
   \*Diskettes?
   \*Other? Specify.
- 3. How many copies of related software publications are provided free with the software?

### Software Maintenance

Once the system is in place, it must be maintained.

- 1. Is software maintenance available?
- la. If yes, is it included in the software price?

°Yearly?

í.

°Other? Specify.

- 2. Is a hardware maintenance contract a prerequisite for software maintenance?
- 3. Does the user pay for software functional upgrades?
- 4. While supported by a maintenance agreement for a typical operating system, what does a user automatically receive from the firm? Specify.
- 5. What must a user do to receive a new version or release of a product to which he is licensed? Specify.
- v. For how many months may a user be covered by a maintenance contract after a new release/version is available if he does not upgrade?
- 7. What types of on-site assistance/maintenance are available? Specify.
- 8. Who would perform this on-site software service?

°Salesman?

°Software engineer?

°Other? Specify.

- 9. Does your firm offer software education?
- 9a. If yes, how is it paid (fee, credit with license, etc.)?

133

### APPENDIX C

and the second second

## AN INTERACTIVE COMPUTER GRAPHICS (ICG) SYSTEM SELECTION CHECKLIST

Once you have determined your functional requirements, you can use the following checklist to write a system specification.

RESOLUTION	I/O, display, software, overall.
ACCURACY	I/O, display, overall.
REPEATABILITY	I/0. ÷
COLOR	I/O, display (number, shades, brightness).
SPEED	I/O, display, software, tape, disk, printer, overall.
MUST INTERFACE WITH	Other systems, software devices.
SOFTWARE	General graphics, applications, languages.
HARDWARE FUNCTIONS	All devices (specific requirements).
THROUGHPUT	Overall, units of work/units of time.
HARD COPY OUTPUT	Needs.
INPUT	Range of data sources.
PHYSICAL ENVIRONMENT	Temperature, humidity, lighting, static, dirt, noise.
DELIVERY TIME	Hardware, software (when the system will be operational).
PERSONNEL CONSTRAINTS	How many operators, with what skills will be available?
RELIABILITY	Maximum down time of the system.
BENCHMARKS	Specific tools to check the system.
COST	Be careful, THE CHEAPEST IS NOT ALWAYS THE BEST!

#### APPENDIX D

#### USEFUL TABLES

1. Tables IX, X, and XI provide performance summaries of the following three major classes of processor components respectively:

a. <u>Microprocessor</u>--a general purpose processor on one to three chips. It contains an arithmetic and logic unit (ALU), an address and data bus (sometimes multiplexed), a built-in instruction set, and the basic control logic.

b. <u>Microcomputer</u>--an all-in-one general purpose processor that is similar to the microprocessor except that the chip contains the main control program in read-only memory (ROM) and possibly a clock oscillator, some input/ output (I/O) capability, and some read-write random access memory (RAM).

c. <u>Bit Slice</u>--an enhanced subsection of the microprocessor's ALU. Besides the slice, which is available in two and four-bit sections, other circuits must be added to provide the basic control logic.

There are, of course, other classes of  $\mu P$  componentsdedicated controller chips that are either custom designed or preprogrammed for specific applications, i.e., calculator chips, floppy-disc controllers, etc.

2. Table XII is a directory of  $\mu Ps$  by primary and alternate vendors.

3. Table XIII shows the vendors that are vying for a share of the bubble-memory business and each is developing its own support components. The fact is that the range of available components has grown faster than anyone expected.

### 4. Table XIV - Operating Systems for $\mu C$

As figure 19 illustrates, an OS is divided into layers; innermost are the essential functions that sustain the  $\mu$ P's very life. Then comes a group of subprograms that permit interaction with the outside world; then a number of utilities designed to fit a  $\mu$ P to its applications.





The "kernel" (lst layer)--typically on-chip ROM that governs CPU activities--is normally inaccessible to the programmer.

Utilities (2nd layer) may be on the  $\mu P$  chip or on separate chips, but they are normally ROM-based as well.

The bulk of most general purpose OS code falls into the outer circle (3rd layer).

Table XIV summarizes a number of commercially available operating systems, some dedicated (e.g., for development systems), some general purpose. The listings are arranged alphabetically.

5. Tables XV(a-i) present briefly the semiconductor technologies currently used in  $\mu Ps$ .

6. Table XVI presents the  $_{\mu}P/_{\mu}C$  families, for a first-pass evaluation.

7. Table XVII gives, by curves, the production volumes for many  $\mu$ Ps and  $\mu$ Cs from 1977 up to 1979.

8. Table XVIII gives the original source  $\mu P/\mu C$  manufacturers and most of the alternate source vendors' addresses, for additional information, if needed.

137

and the second states in the second states in the second states and the second states and the second states and

Microprocessors
<b>General-</b> Purpose
IX.
TABLE

j	linds attend attends reader			fold in mercan statistical	The 402, 402M and 404L are ROM has someone	s corra an tue South the up M har has und the and	event counting capability, as not	As 20 1/0 lines	NUM See wersen er proved	and come in a 40 pm package	ROM less version of Min1400	ROM less version of Misidal, but 128 addies of on chin RAM	ROM less version of MiniS64, chip has 12 4 be	I/U ports	Bautister in the program storage when	Protocology of SABA and and and	recorrection of autou, the se une and BUM here werenes of BCAB that	By and bree, shill the most meaular	(1080 code comprisite, has built in clock	Frevides 13 addressing modes	Sumular to 650% but needs 24 clack	ROM less przytach version of <b>3670, occepts</b> UV EPROMS on top	Autolable un depietion toud nersean	6402 has 12844 on chup RAM, 6406 has no RAM	ROM less version of 6801 single chip wC	Environd 6800 commund act	Has handy during chains capability	Executes 280 enstructions and hes 8085 hes abrections	2014 test version of NYS1050 sunch chile AC	BCM ins unsion of INS1012, 64 bytes of an	Pre compatible but does BCD sublinction	Superseded two-chip version	CHOS equivalent in 20154, and per computers	There are 1.43 and 4-ming versions
aiemijos sseis Deises auto	1			ş	a l	ž		¥ ,	ç ;	1	ž	fa	Yes	;	÷ ,	;	, <u>,</u>	2	Yes	Yes	Tes	Yes	Yes	Ľ,	Yes	Tes	ų	Ĩ.	a l	1	E.	Ne.	8	ŗ
salenjus jauku masi	1			2	*	1	1	2		ł	2	2	2	,	¢ ;	;	¢ ,	ŝ	ş	ŝ	E Ta	ž	ġ	Yes.	Ĕ	ŝ	Yes	ŝ	, a	ŝ	, j	Yes	2	<u>,</u>
maizye imameni yidmazza Asengnat yidmazza	1	1	1	Ter.	Yes	1		£ ;	<u>,</u>	2	Ň	Yes	Yes	ļ	<u> </u>	1	ļ	ž	۲¢,	Yes	ř.	ta	Yes	Tes	Yes	Į.	Yes	Ě	Yes	Yes	Å,	Yes	1	Ē
reguined (V) Vollages	1	2	: 2	=	15 H EJ	15 4 23				•	~	<u>,</u>						5.125	-	•	~	<u>~</u>	~	~	\$	~	<u>~</u>	3 6 12	~	-	\$12-5	1 4 12	•	
erse (biut) Lecreste	91	1	2	3	\$	9		8 3	19		3	3	3	\$	3 9	3	: 3	8	3	\$	\$	\$	\$	3	\$	\$	\$	\$	\$	\$	3	\$	* 1	<u>'</u>
jiene waishs Budhbud	Ko	-	AN N	¥e.	, in the second	Yes		<u> </u>		!	ä	ž	Yes	ž	ļ	ļ	: <u>s</u>	ş	Yes	ş	Ĭa	te t	Yes	Y.C.	¥,	Yes	Yes	ŗ	ž	ž	ş	5	23	!
1/0 ENERGIZ BABIT	ž	Yes	Ņ	ž	Yes	ž	;	9	2	!	ž	Yes	Yes	2	1	Yes	5	ž	ų	Yes	Yes	ş	ž	ş	Yes	ž	192	ž	ž	Ye.	Yes	ž	, <u>,</u>	!
Cabapterity DWA	2	2	-	ł	1	2	ł	2 2	: 4		2	2	¥2		1	1	. 3	ž	Ťa	ż	2	ž	Yn	ž	13	ie I	Yes	¥.	Ves	Yes	Ĭa	ž	Xex A	Ţ
Clock Dr.chup	Yes	2	2	Ves	¥s.	Yes	;	,	¥e:		E.	Yes	Yes			1	2	2	Ves	Yes	2	ş	2	Yes	¥5	,e	, e	R.	ž	Yes	2	ž	ž, 4	
stack registers stack of	•	3412	7.12	-	3	ł			. 3		3	Z	M	N.		ALA	N	M	Z	M	2	2	M	3	3	3	W	3	ł	W	W	3	WN STS	:
Number of internal general-purpose registers	-	2	24	9614	64.4	5.1			3		Į	124.4	2644	3	5 3	-	3	-	-	•	•	3	-	2	82	e ·	•	=	ž	•	•	=	• ~	·
wjsunbtivj <del>sketi</del> Qu-Cyrb	Yes/1	Yev	Yes/1	Yes/2	Yes/J	2		Ver/ 2	Yes/1		Yes/1	Yes/1	Yes/4	Y.e.		1/1	Yes/1	Yes/1	Ves/4	Yes/1	Yes/	Yes/4	Yes/1	Ves/1	Yes/1	Yes/1	Yes/	Yes/5	Yes/1	Yes/2	Yes/1	Yes/1	Yes/A	
sutputitic BCD	2	Yes	Yes	Yes	Yes	Yes		а ў	2		ž	Yes	Yes	Ym	,	, s	Yes.	Yes	Yes	Yes	Yes	Tes .	Yes	a,	Yes	2	Yes	Yes	Yes	2	Yes	,#	2 2	!
comitations 111	¥a	ŧ	-	ž	ş	2	į	9 3		!	ž	<b>\$</b>	Yes	ž	1	1	ž	Yes'	Å.	¥8	Kes	ž	Yes	S.	Aes.	ţ.	¥e	Yes	ž	ž	,ej	2	<u></u>	!
insthuction base shortstifton base (µ5)	N	10.8/21.5	108/21/6	10/20	•	-	4	150	10/20		82/01	10/20	2/4	2/13	1.25/3.75	12 5/37.5	2.5/5	15/3.75	0.0/5.2	5:0/5:0	51/50	5	1/25	s i		<u></u>	22/2	<b>99</b> /2/6/8	1.4/2.8	3/1000	1.92/8.16	250.25	75/10 17/2	
diamana cioca Aseang (1919) yanayan Aseang (1919)	S	11/12	0.74/2	1/20	S	N						0.3/1	5	2/1	0.4/2	0.8/2	5	2.6/2	251	\$ :	5 :	Ş	5	5		5	5 3	12	W	\$	2/2	5	22	
teensatusee susee	2	*	3	22	\$	\$		: 2	3		c :	c	521	9	3	3	*	2	2	*	*	ŧ	~ ;	2 8	2 3		;	Ż	*	<u>×</u>	2	= 1	<b>3</b> 2	
Direct addresang	•	4	4	9261 1926	4	=		. 4	4	4	۶.		4	3	4	3	3	3	3	3 :	1	I	3 :		5		:	ŧ	ŧ	5_	3	<b>3</b> :	14	
(esteringtingtion) More sup	Z	5	5	5	\$	\$		5	5	1	5	5	\$	5	2	5	5	5	5	5	5		5 :	; ;	5		5		5	Ì.	5	5	15	
Process Process	CNOS	<b>Titos</b>	Sour	<b>MOS</b>	SOMM	SOMM	<b>NOM</b>	Sour	SOMM				<b>NIEOS</b>	SOMM	PMOS	PMOS	SORK	SOWN	SOWN	SOMM			SDWN					ŝ	SOMIN	SOMM	SOMIN			
	INCLOSED	Mat	9.09	\$\$\$04"	COPARD	COPARIN	CDPHDAL	95504"	BIN1458			VRCS INC	6651NW	2 chie fs (3850)	8008	1008	8035/8039	8080W	500	10(\$-20H						AND A REAL	MUNCH NO.	Anone	IN SOCIO	6209SW1	A0808 674	APRIL 1		
and the second se	1	3	ł	MC Montempeters	New Second	Intered Secondary	Manual Install	NC Montempoten	Press				Participat	Farchill	Great Internet	]	1	1	1	MOS Technetic			Heleigig			Manual Constructions			Releval Semicoductor	Notenal Semiconductor	MC Knumpelin	1		

A Charles A

tameno	Intended for high speed controllers	BOBO instructions are a subset	Emulates PDP-8 induction set	Has multiply and dende wit.	Control-brended succeprogrammable CPU. con generate CBC hils	Emulates NOVA unstruction set	Erecutes the AOVA milituction set and addresses double the memory of the milital	Emulates MOVA instruction set	Executes NOVA 3 and 4 mstruction sets	Can do deuble word operations	antiputer of the reduced barden (A	Has 24 addressing modes	8 bit bus version of 2086 microprocessor	Nas 32 bit wide internal structure	Architecture intended for data handling	8 bit bus version of dual language (8380/ autive) CPU. Ais internal 16 bit bus	Full 16 but version, effers 8080A and native instruction sets	Lippinded 16 bil version with eight 32 bit rezisters, sin 24 bil rezisters and two 16 bit	registery, can address 16 Maptes	The 9541 reques esternal clack	ROM less version of 9940, with buses	Emulates \$50 minu instructions	Vers trade to Diff (St 1)	function and discle excella Paral node		40 pm series is the 28002. 48 pm, the 2600	
CLOSE BOLLMBLB	¥ ·	¥a.	ň	Ĭa	Yes	č	Ra La	ź	ş	ş	ş		Yes	Ĭ	¥.	1	1	ž	1	les l	Yes.	ş	1	1		ře.	
sazen Juer Historia ereg	2	Ĭa.	ş	S.	2	Ĭn.	Yes	ł	ž	ž	ş	Ş	Yes	Yes	Yes	Yes	¥3	ŝ	1	ŝ	Yes	Ĭa	ļ			ş	
asisas juaudojaaap astanja juutnista	ş	Yes	ž	¥6	Ya	že	Ĭa	2	Yes	Yes	Yet	1	Ves	Yes	Yes	Yes	Yes	Å	141	Yo.	Yes	Yes	ž	: <u>s</u>		Ta	
(A) gaundan sadaring	\$	5	4 10 11	5-5	<u>s</u>	5,10,14,-4,25	1.12.45	Ş		5.1.2	10-1		- <b>s</b>	~	~	<u>.</u>	<u>~</u>	<u>~</u>	1-215	115-215	~	5.125		+12.45		\$	
(suid) 2218 aðensed	8	\$	\$	*	=	\$	\$	\$	\$	\$	9	\$	\$	3	\$	\$	\$	#	1	3	3	3	\$	; \$		<b>ş s</b>	
riene wasses Buidésous	ž	Yes	Ĭe:	<b>,</b>	Ya	ł	Įa,	Yes	Yes	Tes	ž	3	Tes	Yes	ş	ş	Yes	Å,	3	2	ž	2	j =			<b>بة</b> .	
1/0 cucinits avail Specialited memory &	2	Yes	Yes	a l	Yes	Yes	Yes	Yes	Yes	Tes	Yes	1	Yes	Ya	ťa	Yes	Yes	P.	, and	ŝ	Yes	ž	1,5	<b>,</b>		2	
CDESPHIER DWY	2	Yes	ž	ŗ	Yes	Ya	¥1	Ťe:	Ye:	Yes	ž		Yes	Yes	Te	Yes	Yes	Yes	ž	Yes	ž	Yes	,			ž	1
clock Dn-chtp	2	1	Yes	Yes	*	Yes	2	Yes	2	2	2	,	Yes	2	ł	2	2	2	1	Ye.	Ye,	2		: 2		2	
219CF LEEIZEL2 MILLIDEL OL		đ	3	M	æ	W	3	M	3	M	AA4	3	J	M	10-1	RAM	AM	ł	MAG	MM	M	ł		3		3	
fiymber of internal fiymber of internal	-	2		-	7		÷	-	-	3		•	-	*	-	-	•	-		. 2	3	16		3		<b>*</b>	
On-chip Micrupts/levels	1	Yes/1	Yes/1	Yes/B	*	Yes/1	Yes/16	Yes/16	Yes/16	Yes/1	<b>V</b> = 1		Yes/I	Yes/1	Yes/6	Ş,	Į.	Ya	1.27	Tes/A	Yes/4	Yes/16				Ya/I	
sugadic eco	2	le.	2	ž	2	2	2	ź	Ŷ	£	ŝ	5	ŝ	Yes	Yes	Ťa	Yes	Yes	1	2	2	2	1	ş		ş	
and the second sec	2	Yes.	ž	Yes	ş,	Yes	ş	¥e.	Yes.	Yes	ž		, s	Yes	ž	tes	Yes	Yes	ŝ	2	¥\$	Yes	, .	: <u>s</u>		,	
ome notzutzni <sup>S</sup> izzynoi (zaSnofa (za)	0.25	1/5/5	25/25	96/91	0.1/0.2	12/29.5	24/33	1.25/3.5	63/5/	1.19/14	16/48	TIEN O	0.472.8	0.5/NA10	2.5/5	NA10	er VIV	MA10	2/6	3.2/19.6	2.4/50	1E/2	21.75	2.6/300		0:75/90	
fieguency (MH2)/phases Maximum check	5	5	5	2.5/1	1/01	132/2	2/17	12/10	20/19	141	272	5/1	15	5	v	NA 10	KA 10	NAIG	2/2	5	2	5		ž		5	
Banc matriclions basic matriclions	• • •	+951	=	a a	ŧ	¥	2	3	2	153	a	3	8	3	\$	+ 001/82	+001/02	****		5	3	\$	iic	-91		<u>i</u>	
Durstander (moret) Durstander (David	4	3	\$	4	3	ž	3	Ĵ	3	5	3	1	ŝ	N9C	3	3	3	3	3	4	20	â	3	3			
(eses/wather) Mord Suz	<b>8</b> 19	5	12/12	12/12	Javie.	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16	16/16			
Alleytengoog Statution	<u>م</u>	SOMM	CHOS	SOM	g	NINCS	SOMM	z	2	4	N N	SOWN	NUOS	NUOS	<b>SOVN</b>	SOWN	SOM	SOMM	SOM	NWOS	AMCS	/SOMM	- SOMM	SOMM		SOM	
ļ	81130	2002	618	06161	91162-4	10934	509Mm	9446	946	NGAN	CPICONISIA	3808	808	INCESCO.	0060SHI	NS16008	MS16016	NS16032	MN1610	1892-08262MT	2066SW1	00664857581	ND IC	Pascie	Muchengine	50nf/	
	topols	Zihng	Piterie -	1	Annual from formers	]]]	Date Concel	Para	[events	lent	Court Interest	}	1		Retrict Semiconductor	Related Summerica	River Security	Rived Security	Pretices	Trus Instruction	True Independent	Ins Interest	Redets Dela	Western Devel		Ę	

l. Has 8-bit external buses and 16-bit internal buses. 2. With maximum clock. 3. Except clock lines. . Standard TTL or MOS circuits will suffice. 5. Range in bytes. 6. Frame pointer too.

7. Double-precision 16-bit operations available. 8. String search. 9. Clock internally divided by 4 or 6, depending on instruction. 10. Not applicable. 11. 9980 only. (From: Electronic Design 24, Nov. 22, 1979)

÷.,

139

TABLE IX (Continued)

.
E X. All-In-One Processors

1	
- 7	
2	

j	lactudes dispirar dirivers and swatch interface Expanded ROM version	Both the 2200 and 2400 include an 8 bil a/d	currentes and an 10 MM (2 conserves an account processors also come us an "A" version built	can describ deve vacuum fluorescent dasplays Max fruch sauteb satedare	timenal 1/0 version	Andrew CHOS and PMOS and PMOS	tasir handies despise demat	Comes in 54 pin flat package	Designed for mashing machines	Has scientific calculation abouty	All COPs processors include serial 1/0 and	event counting capability Major differences	between models include the 1/4 arrangements	-input caly, butirectional, autout only, etc.	1/0 spinors mouse (co gives years going,	push pull, open diana, and standard (active	device to ground and a pull up to VCC)		•	Well surfed for POS and ECR applications	111 compatible (/0 knes	FUE BANGIES 33 Y BACUMA HUGGESCEN GENE	Living without we get the on the putter!	tow power version of 547 (half the current)	1/0 handles - 35 V vacuum liumescent orme	(MUS version of 547 (45 of the power)	1/0 handles - 35-V vacaum fluorescent drive	CMUS manage - 22 9 Michael Musica (1)	Includes a d convertor with 25 resolution 45	accuracy	Complete all is one controller	Smaller I/U version of 1440	All the processors in the Wallague upony are	12 and 16 we work and an advertised and advertised	and a second and the subscription of the second sec	into a low 2/3 of the state from such the	as the 1405. The CWOS removes can also be	with stratter veloces of at in 10 %.		
Creat software	22	ş ;	2	ų	2	ž	ş	ş	2	:	Ş,	£ ,	£ ,	5 5	, s	ş	¥s.	ş	Ę,	<u>ج</u>	ş ,	e <u>j</u>	: <u>s</u>	ž	ž	j,	£,	<u> </u>	ş		£ ;	<u>e</u> ;	£ 3			. 5	E.	ž		E
werses durawerstone alternation and alter	22	ş ;	: .	ş	ş	×,	Ş	ŝ	â	2	2	1		: :		2	2	ł	2	2		1	2	ŝ	2	ź	2	2 4	2		2	<u>e</u> 1		1		2	2	2	2	2
wassas Durwwiczburd adenduch nigwassy	5 <u>5</u>	ž,	!	×,	la l	ž	5	ĩ	ž	ž	ş,	ξ.	ç ;	<u> </u>		ŝ	¥6	ŝ	ę.	ş ;	ç ;	ç ş	2	ŝ	fes	ş	£ ;	5	ŗ	,	5	ç ;	<u> </u>	3	5	1	ş	۲a ۲a	, s	2
I and maining	4 2	, ș	2	Yes	2	1	ş	fa	2	2	ş	Ξ,	٤,	g j		ş	la I	ŝ	Ş,	ę ;	ç ;	ç <u>ş</u>	2	ž	Yes	Ę	£ ;	e g	Ĭes			: ;	ç ş	3	1	ļ	ġ	ş	2	5
(A) filindas Sidayan		•	•	10 to 20	9	- 10/ + 5	2-	2	-15	•	(5 le 9 5			24 16 63	454653	451035	24463	190153		2 9		2	2	7	ę,	÷ ;		: -	9-			~ ~	~ ~	, .,			21	1246	425 46	12646
(Dib 8-00) backs and	33	<b>8</b> 9	;	2	22	3	3	*	11/24	=	× 1	e ;	: :	: :	2	2	2	3:		* 5	* 3	:3	: 2	3	<b>2</b> :	<b>*</b> 1	2 2	: 2	2		* *	8 2	: 2	: 3	3		\$	\$	3:	2
CHASHS THERE	22	1	:	2	2	a a	Į.	in the second se	<u>T</u>	Ş,	<u>و</u>	ę ,	ę ;	e ș	2	, a	a l	, E	E .	1		1 2	2	2	2	2	2 4	1 2	2		2 4	. ;		ž	1		ŗ	ž	, i	,
sour 0/1 p source of	22	21		1	2	R	=	3	2	=	2 ;	2 5	3 5	2 2	2	2	2	2	R ;	с x	ç z	: 2	2	2	\$	s ;	3 2		2	5	R :		: 9		3	1	z	8	2	2
anter anter Anter anter	11	33		1	3	A	MM	3	~	-	3			3	2k	M	RM	3			1	3	RAM	3	2	3		ł	W	1			3	2+ BAM	M	3	2+RAM	3	3	3
Standi Julian Sultani an	33	~ ~		-	1	-	3	3	~	-	~ 1	•••	• •		-	-	-	-	•	• •	• •	. ~	-	-	-		• •	-	~	•	• •		. ~	-	~	~	~	~	~	~
<b>maturb</b> @ <b>\phane</b> gu Cuib	Yes/I			Ver/1	1	Yes/2	Yes/2	Yes/2	~	Yes :	2 3			16/1	ş	â	ł	ž		745/2	Yes/1	Ye.	Yes/	Yes/1	Yes/	16/1		1/12/	Yes/1		/24		Ň	Yes/2	Yey	Yes/1	Yes/2	Yex/1	1	ş
946-344 606	Ka Yas	5,	}	Yes	Yes	¥a.	Yes	ş	~ ,	, <b>2</b>	5,	9 3		<u></u>	ţ.	Yes	ž	ş,	£ ,	<u> </u>	Ş	2	ŝ	¥e:	<u>2</u>	¥,	9 3	r.	ž	3	9	1	1	-	4	2	1	2	2	2
P(TupBudg) LL	ţĮ	<b>z</b> 1		1	2	No/Ver	1	2	~ ;	<b>z</b> ,	ŧ,		: 5	2	fa	ž	ş	Ş ;	<u>,</u>	: ;	: ș	Ya.	745	5	ž,	ç ;	, <u>,</u>	ž	ş	3		2	2	2	1	2	2	ž	ş,	E.
and marked and a first and	55	55		6W/20	2	2	2	2	-				2/31	16/21	5	16/32	16/32		96/00		5	15/3	4 2/8	11/22			5	Ş	15/3	14/70	10/20	10/20	07/01	10/20	15/30	15/30	15/30	273	3	ž
ting ga tipe	ž ž	g ;		Ϋ́ς.	Ę	r.	Yes	ş	£ :	2,	£ ;	5	ļ	2	Yes	Yes	2	5		1	2	Š,	Yes	, e	ş ;	£ ;	1	Ţ.	Yes	,	ļ	Yes	Yes	Ťes	Ves	Yes	Yes	Ter.	2	=
(7)41) Kaussang (2010 Kaussang	<u>.</u>			Ş	2	788/500	ă	2	2	3	8 2		2	2	1000	2	2			3	1	87	914	3	ŝ :	3	1	3	9	W.	8	8	8	DOE	2	8	g	<u>ş</u>	<u></u>	
geleic subjitigene geleuter be	33	8 9		3	z	z	5	3	-	2 :	<b>3</b> 9	2	; 5	: \$	\$	\$	\$	ç :	; ;	2 2	: 2	3	3	3		2 3	: 3	3	5	×	: 2	: \$	3	z	2	3	£	z	9 :	]
unsatin Louing Au ya	11	5 3		Yes	2	1	2	2	-	5,	<u>s</u> 1	1	1	1	ş	Yes	5	¥ ;	! ;	2	1	ł	ž	<b>8</b>	2,	<u>c</u> 4	2	£	7	1	1	2	2	1	ŧ	1	1	ł	1	•
(Water and ACB) (Water)	1024×4			1024×8	512×10	1024×16	2048×10	2048×10	N/N			1074×1	1024×8	1024×8	1024×8	1024×8	1024×8		1420×10	2000	2000 × 8	2600×8	1003 - 4		1×0001		1000×8	1000×8	1×0001	1074×0	7674	12215	512-4	2046×8	1024×8	<b>Jiuxt</b>	2048×8	1024×8	52×1	
jit. An circle ANK	RX4 RX4	128×4		64×4	32×4	<b>8</b> 0×4	163×4	16×4	1	27.76		E X A	61×1	1×1	64×4	64×4	64×4				9 × 9	\$6×6	HXH				1XX	12×4	ž	EAV A	32×4	19×9	16×4	28×4	HX4	1×2	121×4	žX.		Į
( phar : Terp) Bry a bin gatty	55	55		\$	5	<b>81.</b>	5	Ş			; ;	5	5	\$	5	5	\$ :			5	5	\$	\$	5 :	53	5	5	\$	\$	5	5	5	\$	3	5	3	5	\$ :	31	]
Mapping 14 Mathing	South			S	Sond	CHOS	5044	Soar			Ne of	N NCS	NVOS	CNOS	NUCS	SOM			SUNA	Sen	Puros	CNOS	SOAL				205	CMOS	NCS N	NVOS	1403	SCMM.	SOVA	SOMIN	SOM	Son	SOAL	50		j
Ĩ	2023			24	NINCSA?		NUCSAL	HINCSAS I	1 22 1 20 1 20 1 20 1 20 1 20 1 20 1 20			021430	COFEZO	COPAZSC	127480	COPULIN	C07421C		27.07	WSON -	1055	P0650	(1500	1 YNGA	2003	1550	Kich	2590	15504	CON LINE	2011462	ECHINA .	PEPINA	Solva	CENINA	ANI 432	STIM	057 ( Ma		
IJ	Į				Nuch.																																			

140

- Ailti

5		A "I SUSSE LAND DESCRIPTION IN UNIVER IN	The state of the second states of the	register, 48 4/0 team are bedrochenel and the	Charles and there is prover goods many in the party in the party sectors.	Combration BOM/BAM///O and the		Gan a MSI M IN ANNI A	1.0 industrs securi channel	Sultant cumption with 7	Primarity wild be beliand depine	Has high speed counter	Luger BOW Pare MMA	ton punct written of N	Espended KUM eerses	Reduced 1/0 version of N	Aude from the tubols services a 35 Y victure	Numericant dual worken a medathe (145	10/0/12/01	CVUS mercian at 1415-1878	With Late Version and and All 1170/12701	and is and consultation with 1145-1000	Dedicated must ber crutecher	Dedicated CB PLL contractor	Bedicking microacters and canimilar	Brdicates and and timer/controled	VE drive version available (1478) and 15 V	supply model coming	VE drive version sublicite (1670) and 15 V	Junghy model comune	Reduced POM and 1/0 version of TMS 1000	telended for dedicated controllers	Designed for begtaurd'd spiny interforme	Part huids Bull numbers	uniting by frace marker of more must	Sent to Maile 202 M has to stand.	14	Mimmal 1:0 family members	32 ptogrammathe 1,10 knes	Reducted 1.0 version of 1656	lactersed memory perjude	Minimut 1/0 CPU unsue	Contains the ard concerter channels	toti has a fow and 8741 a W EPROM	BITE NO UN CARDIN	feliged memory return of 1040	La da aquineal in Intel 2041, <b>BANNEL M</b>	CHOS manatest to teld M.42 (N.42 and	supates 50 mm	No. 16 bet pang daman Davides the memory capacity of MM.
Antonias vers Burness Aury	5	2	5	£ ,	5	ş		ş	ŝ	Ę	ç	£	2	ş	2	ş	2			ş	Ę	•	ALC: N	N/A1	1. Y.	14.4	ş		ş		5	5	Ę į	ç ;	2	ş		2	2	2	2	F	g	E.	£ ,	, s	ş	ç		<u>.</u>
201001 20-000000000 20000000000000000000000000	12	2	1		ŧ	2		2	ł	ł	1	2	2	2	2	2	2			1	1		-	14/1	*/A*	1.01	2		1		2	2	F,	F ;	2	ž		ş	ě	ş	ŗ	ş	ş	5	E,	E,	ę	ş		2 5
uside Donatifiers adtRouts diffuorig	1	£ ,	ç,	£ ,	2	Ťe.	_	lei l	Ves.	ş	ş	ş	ş	ž	ş	ş	ş			ş	Yes		N/A <sup>1</sup>	. N/N	1V.	N/N	ş		ş	,	5	£,	5	£ ;		ž		ş	ŝ	ç	ç	č	ş	Ę,	£ ,	£ ;	ç	,		ž ž
pillet wastes Buddipiona	E.	, i	E .	£ ;	ç	Ya		ş	ş	ŗ	ş	ş	ş	ş	Ę	ş	Ę			ş	e,		-	N/A	1.1.1	N/N	Ę		ŝ		F :	ę,	5	ç ,		že		ş	ş	ş	Į	ş	ş	£ .	5	£ ;	ç	Tes.		z z
(g) plankja sikujag	1751	•••	^ `	•	•	51-3+/(1-		-11/+512	-15/+110	-15/ +5 - 10	-15/ + 5 - 10	-15/+510	01-3+781-	561	01-3+71-	-15/+5-10	9 é 15			346	3 15		2	2	2	2	-		-	;	5 × 1	~ .	• :	2.	•	•	'	~	~	~	-	~	•	•	•••		2	199		
(Bib Ball) product and	3				5	3		3	3	¥	<b>¥</b> .	3	3	8	3	2				2	3		2	z		\$	2		*	-	<b>R</b> 1		3 1	3 9	;	\$		2	<b>\$</b> .	z	=	z	8 :					=		**
50000 (1000) (1000)	1-	<u>,</u>	e ;		1	ş	-	ş	ş	,	a a	ş	ŝ	ş	ş	ş	ž		_	ş	ş			ł	NV.		ş	_	ş		5		9 4	1 5		Yes		ä	ş	ş	ş		2	<b>P</b> .	£ ;			, a		<u> </u>
Sam 0-1 Rumber ge	= ;	: ;	: :	•	;	12+	_	12+	=	=	=	<b>R</b> :	3	7	×	2	S S		-	2/22	3728	~~	1	ŝ	, v	÷.	2	1	2		; 2	. :	: :	: 2		ñ		-	R	-	2	2	2	::		. =	:	8	1	* 2
tatu itu puntu Matak Matak	404 - 2					-		-				1	I WW	NVII +	-	N.	Ĩ			11:	I HI			N/N		1/N	1		1					1		3		ž	AN	3					1	-		3	1	11
synat decies decimating						~		~	~	~	_	-	_	-	-	_	-		-	-	-	·	i Vi	Š	Į.	ŝ	-		-					1		-		~	~	~	~	1	1		• •	• •		-		11
94 54/54/14 49 (14 64						Yes/I		2	Į,	Ň		TAVI	Nev.	ž	λ«Κ	Į.	£			1	1		2	2	2	£	1			á						fes/4		ž	(a/)	2	2	2	5				;	S.		Ň
ادیون مارد ۱۹۹۵ -	2,	,	; ;		!	a l	;	ş	Ş,	ē,	5	5	ě	r,	ş	,	ŝ			ş	ş		ŗ	ş	3	ş	ş		ŗ	;	-		,	1		ž		ş	ŗ	2	Ę	2	¥ ;	c ;	٩,			ş	,	5 5
ing state and state	2	]	• •			2		1	2	2		5	3		5	ş	ž			ş	ž		ş	ŗ	3	ļ	Į		Į	ļ	1			1		Yes		Ę	ŗ	5	2	Ę.	2	ç ,	£ ,	1	;	ľa		<u>e</u> <u>e</u>
an ( and a funder ( and ) and ( and ) and ( and ( a funder ( a fun	(1) (1)		1			ž		ŝ									15/15			3	15/15		1213	ivis	12/15	15/15				16/16	-	11.160	22/12.5	1/65		1/65		5	\$ 1	5						5	-	2.5/5		
درعدو وب دوسک	5.		,	ļ		1	1	5	<b>,</b>	5	<b>,</b>	£,	ž	5	2	ž	2			2	Yes		ş	, a	Ţ	ŗ	1	į	1	ļ		ļ		1		Ë			8	<b>\$</b>		Ş,	;	,	,	ļ		Ň	;	: <u>s</u>
(tys) (summing t) p ununty	3	1	1	1		200/400	Tas clucks				1001	<b>\</b>	ž	ì	ž		\$			ē	ş		ŧ	\$	ş	ŧ	3	-	Ā	1		Į	2	3				8	3							6000 (SY)		6006 (SV)	-	1 201
parc all: come graps a	k a	1	1	5		8	;	3	8 :			<b>R</b> :			8 :	8	3			3	\$		\$	3	3	3	÷		Ŧ	4	; -		; 2	ź		ŝ	1	R :	3	2 :	R ;	2 1	2	2 3	: 3	: :		*		ż
1941-1949 Auruse 942-183	2.5	! 3	1	: <u>s</u>	1	ž	,	Ę					i	ſ	ſ		2			2	2		4	2	2	2	2	i	1	4	1	1		2		2		2	2	2	2	2 :	ŧ ;			: 2	!.	ž		£ 3
6000 876 ( 400)) De tuñ 800)	8 - 8 - 8 2000 - 8	ADVA.	2 AMAX	X SOS		•		•			Ň				ž					1024/1	244		1024×1			X TX	X	611-10	1×21c	\$12×8	256.2	2.42	12×10	248×8		8×9601	:	2		21 x 21C				No.	2wax5	1924×8		1026×8	2044.4	8×900
946 1946 998 1946			122	XXX		•		•	ž.				ž	Ĭ		ž	ž	_		7×3	128×1		ŝ	Ĭ	1.1	12174	*×12		141	64.4			32.44	ŝ		8×9									178.6	8×8		ŝ		N. N
1 ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) (	::		5	5		5		5	5 3	5	; :		5		5	3	Ş			5	\$		5	3	5	\$	Ş		5	1	3	5	5	3	_	2									5	5		3		5
ditanen in se		NM0	ANON.	South		SOIL								2		Z				Geos	ş		Ĩ	Ē	Ê	5	E	-	1	MEDA	ACC.	SOUM	FINOS	A ROS		Some	1		5				C. C	YUN	SOM	Since		500	CALOR .	1 Million
<u>F</u>	C() ( 1978)	AN: NA	100	1951		154		Ē					1.9/6.3			Supa	Since Inco			THIS IGUAL	Tins Lice		100 1010	27015.01	11115044	<b>Witsat</b>		THEFT		Day 12 miles	13446	11411	2/11	F MK 10	_	13676	-						1010/1000	AAA. 8748	S	D/CA1		MC44/87C48		Ĩ
							_									•														•	ler Auto		Malmo hald	1 autors							]	ŀ		-		ļ			-	-

TABLE X (Continued)

· 32.4

March III (1994) March III (1994)   Marc
1100 15/25 49×25 152×24

4. Not available. 3. User defined. 1. Not applicable. 2. External 8 bits, internally 16 bits. (From: Electronic Design, 24, Nov. 22, 1979)

TABLE X (Continued)

TABLE XI

# Bit-Slice Families

715		sectivology Process	had UJA namun	(slig) əzis picm (ital)	ALU instructions Mumber of	Can ALU do 503 antimetic?	Meximum ALU clock rate (MHz)	Ceneral-purpose	ALU package size (DIP pins)	sequencer number Microprogram	Number of address bits	Maximum sequencer clock rate (MH2)	sequencer commands Number of	stach size Sequencer	sise (DIP pins) Sequence pechage	IIL-compatible?	Voltages	aldelieve maiste	eventions and		
, ,	Series Series							-		-		-				_					Comments
Mumor	<b>1</b> 2300	a sin	¥1062	-	2	\$	16.67	9	3	11/6062	-	9	22	txt	28/20	Yes	5	Yes	Xa:	H Say	as midest number of second sources
Device	~	SITL	2903	-	x	2	2	2	\$	2910	12	2	9	5×12	4	Yes	~	Yes	Yes	Yes	I'd has nine more instructions than 2901
Farchid	) Macrolegic	CHOS STILL	BOLTE / SOME	•	3	2	2		24	3406	-	9	•	16×4	24		 5	Į.	Yes	<u> </u>	ktuainz muliupiy and divide. MOS version (34705) operates at 2 MMz.
•=	F100226	<u>ਤ</u>	F100220	•	2	Å.	3	-	3	100224	N/A <sup>2</sup>	N/N2	2W/W	24/W	N/N	N/A2	-4.2 to -5.7	1 6/61	616	0 086	nly 8 bil slice, sub-ns instructions
Malarda	1000	g	1080	•	*8	Yes	2	•	\$	10801	•	2	9	4×4	\$	2	-252	ζa	Ke.	Yes	astest 4-bit since available
Sugnetics	3006	a sm	2005	~	\$	£	2	H	38	1000	•	+01	=	•	\$	ş		Yes	Yes	Yes	nty 2-bit ALU available
1	. SBP-04004	z	SBP-0400	•	512	ž	\$	2	\$	200521	-	2	3	1×1	50	ŝ	Current	Yes	No	No H	as pipeline register
	100-085	E	S&P-0401	-	212	ž	\$	2	\$	745482	•	2	3	1×4	2	ş	Current	Yes	٩ الا	ة يو	oes not have popeline register
	IBYSHLING		2481	-	24.780	tio	2	0	4	745482	-	2	3	4×4	2	is i	s	Yes	2	<u>رم</u>	ery Heuble instruction set
		15in	18r51																		
																ļ		ł			

(From: Electronic Design 24, Nov. 22, 1979) 2. Not available. 1. Leadless carrier.

# TABLE XII

# Processor Alternate Source Directory

ł

U

Generic type number	Data word size (bits)	Technology	Original source	Alternate sources
1600. 1610	16	NMOS	General Instrument	EM&M Semiconductor
1650, 1655	8	NMUS	General Instrument	EM&M Semiconductor
1802. 1803	8	CMOS	HÇA	Scientific
1872	4	PMOS	Western Digital	None
2650	8	NMOS	Signetics	Advanced Memory Systems National Semiconductor
2900	4	STTL	Advanced Micro Devices	Fairchild, Monolithic Memories, Motorola, National Semiconductor, Raytheon, Sescosem, Signetics
3000	2	STTL	Intel	Signetics
3850	8	NMOS	Fairchild	Mostek, Motorola
3859	8	NMOS	Fairchild	Discontinued
3870	8	NMOS	Mostek	Fairchild, Motorola
4040/4004	4	PMOS	Intel	National Semiconductor
5701/6701	4	STTL	Monolithic Memories	ITT Semiconductor
6100	12	CMOS	Intersil	Harris Semiconductor **
6400	8	NMOS	Motorola	None
6500	8	NMOS	MOS Technology	Rockwell, Synertek
6800, 68A00. 68B00	8	NMOS	Motorola	American Microsystems. <sup># #</sup> Fairchild, Fujitsu, Hitachi Sescosem
68000	16	NMOS	Motorola	AMT 38 Frieshild
6801	8	NMOS	Motorola	None*
6802	8	NMOS	Motorola	None
6809	8	NMOS	Motorola	None*
7150	4	PMOS	ITT Semiconductor	None
8000	8	PMOS	General Instrument	AEG. SGS-ATES
8008	8	PMOS	Intel	None
8035, 8048, 8748	8	NMOS	Intel	Advanced Micro Devices. RCA, Intersit, MataSignetics", NEC"
8041. 8741	8	NMOS	Intel	None
8080A	8	NMOS	Intel	Advanced Micro Devices, NEC, National Semi- conductor, Signetics, Texas Instruments
8085	8	NMOS	Intei	Advanced Micro Devices.
8084	16	NMOS	Intel	NEC , RCA
8900	16	PMOS	National Semiconductor	None
9002	8	NMOS	Electronic Arrays	Discontinued
9080	8	NMOS	Advanced Micro Devices	Actually an alternate source for 8080
9405. 34705 9440	4 16	STTL/CMOS	Fairchild (Macrologic) Fairchild	Signetics None

A COLORINA

# TABLE XII (Continued)

Generic type number	Data word size (bits)	Technology	Original source	Alternate sources
9900	16	I²L/NMOS	Texas Instruments	مه American Microsystems (NMOS version) م <sup>عهر مع</sup>
9940	16	NMOS	Texas Instruments	None*
9980	16	NMOS	Texas Instruments	None
10800	4	ECL	Motorola	None
14500	i	CMOS	Motorola	None
8X300	8	STTL	Signetics	None
745481	4	STTL	Texas Instruments	None
100 K	8	ECL	Fairchild	None*
COPS	4	PMOS	National Semiconductor	None
F8	a	NMOS	Fairchild	See 3850
FLOOL	16	Bipolar	Ferranti	None
MP-4. 8. 16	4. 8. 16	PMOS	National Semiconductor	Rockwell
Macrologic	4	STTL/CMOS	Fairchild	See 9405. 34705
MCP-1600/WD-16	16	NMOS	Western Digital	None
Micromachine	8	NMOS	Fairchild	See 3859
mN601	16	NMOS	Data General	None
MN1400	4	NMOS	Panasonic	None
MN1610	16	NMOS	Panafacom	None
PACE	16	PMOS	National Semiconductor	Rockwell
PPS-4	4	PMOS	Rockweil	National Semiconductor
PPS-4/1	4	PMOS	Rockwell	None
PPS-4/2	4	PMOS	Rockwell	None
PPS-8	8	PMOS	Rockwell	National Semiconductor
PPS-8/2	8	PMOS	Rockwell	None
\$2000	4	NMOS	American Microsystems	None
SBA	1	NMOS	General Instrument	None
SBP0400A/	4	12	Texas Instruments	None
0401A	1			
SC/MP, SC/MPII	8	NMOS/PMOS	National Semiconductor	Rockwell, Signetics, Western Digital
SMS-300	8	STTL	Scientific Microsystems	Signetics
SX200	4	PMOS	Essex International	None
T3190	12	PMOS, NMOS	Toshiba	None
T3444	4	NMOS	Toshiba	None
T3472	4	NMOS	Toshiba	None
TMS1000, 1100, 1200, 1300	4	CMOS, NMOS or PMOS	Texas Instruments	Motorola (for CMOS version)
UCOM 42	4	PMOS	NEC	None
uCOM 43, 44.	4	PMOS	NEC	None
<b>Z8</b>	8	NMOS	Zilog	None*_*
280	8	NMOS.	Zilog	Mostek. Sharp, NEC . \$6\$
28000	16	NMOS	Ziton	Rockwell ##

\* This product is still in development. \*\* Licensed. (From: "Microprocessor Data Manual," D. Bursky, 1978)

145

-----

Manufacturer	Model No.	Net capacity	Organization	Package	Sample availability	Sample cost	Special support ICs
Fujitsu	FBM0102	64 kbits	Major/minor	18-pin	Stock	\$100'	In development
	FBM0201	64 kbits	Serial/parallel single-loop	18-pin	Stock	\$1003	in development
	FBM0301	256 kbits	Major/minor block-replicate	16-pin	Stock	\$500	In development
Hitachi	нзм	64 kbits					
	нэм	256 kbits		No	information avai	lable	
	HBM	1 Mbit		•			
intel	7:10	1 Mbit	Major/minor block-replicate	Leadless	30 days	\$2000	Yes
Motorolal	R3M256	256 kbits	Major/minor block-replicate	18-pin	Late 1980	-	In development
National Semi- conductor	NBM2256	256 kbits	Major/minor block-replicate	1 <b>6</b> -pin	Early 1980		in development Early 1930
Plessey	P8064/S1	64 kbits	Serial shift register (no loop)	12-pin	Stock	4	In development
	P8256	256 kbits	Major/minor block-replicate	18-pin	Mid-1980	-	in development
Rockwell	R3M256	256 kbits	Major/minor block-replicate	18-pin	Stock	\$500	In development
		1 Mbit	Major/minor block-replicate	18-pin	Late 1979	-	in development
Siemens	RBM256	256 kbits	Major/minor block-replicate	18-pin	Late 1980	_	in development
Texas Instruments	TIE0203	92 kbits	Major/minor	14-pin	Stock	\$100	Yes
	T180303	254 kbits	Major/minor block-replicate	18-pin²	Stock	\$500	Yes
	TIB0250	256 kbits	Major/minor block-replicate	24-pin	2nd gtr 1980	-	2nd atr 1980
	TIB0500	512 kbits	Major/minor block-replicate	24-pin	4th qtr 1979	\$21005	2nd atr 1930
	TIB1000	I Mbit (512 k × 2)	Major/minor block-replicate	24-pin	4th qtr 1979	\$31005	2nd qtr 1930

TABLE XIII. Bubble-Memory Devices

Alternate source for Rockwell ROM256
70-mil centers (an 18-pin, 100-mil version is also available)
100-unit quantities.

4. Only board-level products will be available (\$2997, 64 Kbytes, SBC-80 compatible) 5. Board-level evaluation subsystems with support IC's.

(From: Electronic Design, 24, Nov. 22, 1979)

TABLE XIV

**Operating Systems for Microcomputers** 

				File tytem territor to 2005 PULKC	· Durate ward. Bouble Sonsity	future 11 Mb			Ca anostado -	Application immune exercipment 1 wor typical 3+	Rag & paper tape user delived devices Funitime only	1 MP OS compatible MOS				I faild delinable P 16+8 X48 tb with setional bank	select P(64K BNB) *(mair BNB) each)	In development 1918 P.A.silable in ROM			1 Proce types	B CR16 man B Contraints
	Comment		UCED computation	Fuil DMA ERCDC ceding	BCD specialism, print, CRT macros, business examled	MT 809 in similar to AT 68 but tor 6809.	Communication		supports 3 5 B m. Norpy date and 8 m. Winchester hard date	Full development & rumume Development en disk or diskette	tioned a Norda or Norw Ectipse minus	Competitive with DOSI RUOS funitime anig	PROM Leaved OS for Move loaved Develop on APPrOS. DOS. A05	Competible with PDP-1	11/39 mituction set	Much add sollier av. Irum outside veriders fitasic Fortran, Cotod.	COMMUNAJIUNS BUS Neus Applic etc 1 MPMM In uumarid computibile Rom C.P.M	Sumule, computible with Bugbook fulurials	buter.active selectored propractionance dataon.	ary Meachure target		
	Prisester A		ins printer.	Tape drue <sup>2</sup> . 800 bPI MRA. 1600 bPI PF	Inten CRF (25 Lines × 80) 2 ser. channets 2 ser. channets 1 par prir pert	32 ker chan man 512 divorate		CRT terminal.	Dol-matrix pr. Character pr. a/d d/a	6 anyric ports. 8 hire pri. 4 eshis	18 async perts 2 him pel / 8 dists'	3 argent ports 1 here prist 1 dish	l aver pert 1 per (32 bely	CRT, hime primiter M.E. 488 Bus.	RT clock 4 ser channels	User defined consule reader.	i und	1 ser R5232. 20 mAri 10 9600) 1 cavelle: NC (300 1200)	CR1	Printer	CRI NAM DMAN	a boot a
	Campinets		Parcel	•	Cete	Pascal Basic	M/M	Fartran IV Ration Cobel	Fortran IV Rattor Cobol	MP/Fartian	Forman W DGA	N WANG	WA	Fertion IV Bosic + 2	Fection IV Basic+2	-	•					Pucel
	belorge over 1		<b>B</b>		Basic	P.code	MA	Entended & Structured Base	Multurer Bauc	MA	Ext. Bus. multurer Basic	MA	NA .	Mine Boste Facal-11	Mine Base Facal-11	•	-	24 Benic <sup>0</sup> Bin Basic	1 arth			Resic
ľ	Assembles	000 22000	900 01. 05. 09 900 9940 5.000 2200	-	280 miles	6800.09. 200 8080.6502 9900.110711	6099	790	942	Mucro	Macro	Macro	Macro <sup>1</sup>	PAL-11 Macro-11	PAL-II Macro-II	10801	łogow	<b>Š</b>	8-5-0804	7 HD 66UX).1802	-0407-76, 280 6800-02	
ſ	#		5	V/N	1	0 4 Mbr	•	BO MAN (B data)	10 M	100 Mb	44 Mm (4 JISAS)	(Nam 1)	ne dest support	21 Mu (4 disha)	anti Mus Yus	120 610	128 MB	₹.	-			1
ſ	l	ŧ	1	1	1 3	91 95 26 19	4 8	512 Hb	612 H	2 3	4 5	1	9 3	9 3	3.4 M	1 3	<b>8</b> 3	3	1	24	41 38	1 3
	11	ŀ	2	ž	ř.	Yes	Yes	2	9¥	Yes	Yes	N.	Yes	Yes	Yes	ž	Xes X	<u>;</u>	rey	**A		Yer
Γ	Į		2	2	-	94	ž	9N No	ł	MA M	42 H	¥-X	4	Yes	Yes	V/N	X.X	2	472	N/A	ž	Yei
ſ	Ħ		2 2	2	1	•	S.	ž	¥,	ę,	Ves	Yes	Ves	ž	Yes	Ŷ	say	ł	X an	144	2	ł
ħ	k		1	-	ž	-	-	-	-	MA	M.M.	WA	M.M.	-	-	WN	WA		W/V	NN	-	
	ļ	ŀ		-	-	<u>-</u>	2	-	~	à	9	i	3	•	2	V.N	2	-	-	-	-	•
		ŀ	-	à	٤	-	æ	13	<b>1</b> 1	•	*	•	•	•	•	î.	2	د: ¥	a T		-	•
	1		3 <b>5</b> 5	2.80	382			2804	YOR7	MINGOL.	MN601.	MINGOI. 602	1009 602	154-11/23	MA	N.N	N/A	€	940808 11751	230 660	60. LBUD UATON	280
	111	Contraction of the second	Conne. To Paori. TCK BOOR	MC2 Linty	¥. S¥	, abred	5000	and and	System Three		109MM	anthe a	A Nou a	13411112	(2/11-151	BORD/280 bourds	BOSOV/BO Lucida		tine, UCC	Zilling, RCA	B080-5-6.	280 boards
ſ	1		Parcel	150MM	Concert		MT 809	CDCS	136 153	SOVAN	500	RTOS	MBC/M	NT-11	WIT-258	CPAM 20	N-LAN	9	Pulyforth	Mkre Forth	SCON	SOGN
	Ì		Cues CA 9051	Men Cers. 2150 invalue Pury Super	Cancedo Bela 6100 78th St. 51. Grand Rupuh, M. 1950	Control Systems, 1317 Control Ave. Kantad		Cremers Int. 280 Bernudo Ans.	frore	Puts Concret. No. 9. WMoro, M. 01581	•			3	Name and	Polyner Printer Polyner 579	Syral Syral	Cal Barray Barray Di Faul St. Daving Ci Daving Ci	Feeth Mc. B15 Munhur-	But h CA	Gendlad/future- data 6151 W.	See 1124 Los

17 . 4.4364

TABLE XIV (Continued)

l.

		1				3	1	1		-		1 days	iorted Inneventer				
Ţ	14 martin	A		1		5	Ĭ	ł	Ī		1	Assemblers	briter protors	Compliant		Contract	
	00100	01 145 Mag	Ng Ng	•	-	2	•	*	**	ŝ	10 10 10 10 10 10 10 10 10 10 10 10 10 1	6050, 790, 1802 etc.	Part -	ferten	M. CAT (20 = 00) 3 ser post. 1 pur KEE 446 (24 bit)	Hardware Naobing	- Opnensk 1 Wesh 6901
Annual Carp	88	81.1M	q	F.R.M	•	•	٨u	£	49,	1	128 440	001 000	XYBase	CPAN	CRT winter	On such have all	· Depends on
	16-50	10.2.00	082	F.R.H.	-	•	¥	ž	ŗ	1	1 2 1	6080, 280	KYBosec	MP/M Classic Others	2 510 4 P10	minitisca. 05-61 alta main precessor	AUNA MUP NO MUNA MUS POS BI nos 4 80
<u> </u>	0000	m.2.040	8	=	M	•	Z	2	ş,	3	MA	-	-	-	CRT, primer 2 SID, 4 PID	On-card 32 channel A/D converter	0
	NTOS-11	11-104	19-11	FL6	2	-	Acts	ž	ş	2	Ng	MA	NIA	. AM			1 Only source Language another
	MTOS-80	080	800, 8085 260	Pi.e	2	-		X.	ş	23	ž	MA	<b>N</b> /N	MA	CRF, floppy. line pruiter.	Email suo. fast intersuot resonne.	Fource and shared forms available
	MTCS 64	993 993 993	1089 0089	н 'ч	=	-	a l	ž	ž	23	ŝ	NA	NVA	MA	11 TOOLOOLON	untimited number of tashs	Dynamic antuiger included
	MICS M	0066	BCB6, BOBB	2.5	2	2	N.	ца Ха	ş	1	1515 Link	N/N	NA	WA			
	15151	Intellec	8048. 80. 85. 66	4	-	N.N.	2	ž	ş	1	*	8048 80.85. 86.89	Bess BO Cobol BO	PLM 80 PLM 80	Cart.	Ruiz-BO supports moth board druk (floppy & hard fypes),	
1	RALK BO	\$8C-80/m	2009/0002	æ	-	N.	<b>8</b>	ł	¥#,		90 MB	NA	Fortran Base	N/N		uro contreller	
Meca Mer Systems, 1650 Humanue, Breca RY 14550	SOCIAL	1-540	987	-	-	-	1	2	ţ	1 3	120 Mb (1 041)	ZBO macre	Baric	Parcel	CRT (16 = 64), perilary!, F or H 6mh	Pascal re-entrem A NO: Lobie Interface In OS Through Isbuery	1) the printer. I cher. printer
Microbial 711 Services Bird. Alte- models Sprungs FL 32701	DIC OF	DEC POP Luminy	11-004	<b>K</b>	3		Yes	141	¥,	3121	191 5	- Ma	mguage process rading system ci Microbal		Serial devices. CRI's multiple attach com- mand	10-100 times laster hian DEC Basic er Datol (na computer ben required)	14 46 29 707-11. 12-11/23
Systems Systems First Sylves		sec territy		£	-	•	2 ·	2	ž	4 2	•		Bå.		CRT, printer, VO channel (user defined)	Low cast, easy to the corporate into user system; LO expansion capability	t Some routines in MC Depends on SWC op 1.3 Hoppy disks
Muth Certe 1715 W County ML Controller	00 04-14	92 20	8		-	-	2	2	ş	4 5	\$ ~	11 7 0007	Parcel	Ferren	Terminal. Ince printle. Mispry disk.	05 for Moster Nakes development system	Cossassembler Macroassembler Filted DUS only
	SOODA-1	ZIID farmer	887	-	-	-	ž	2	ž	2 3	3 ~	1907	Party	fortion	PHOM Progra	Development OS for \$10- 280 Lased MD boards	
	100 DOS	(inne) 082	2	Ĩ.	-	-	ž	2	Yes	1 3	4 MP		Beuc		Salerit 700	Production OS for STD- 200 based MD boards	
	MITE-BO	280 (ame	287	2	-	-	ž	2	ş	1 3	4					Munissiane 05 for 510- 280 Lased MU brands	
	DOF-BO ASMB 00	ZBO farmel	2	e.	-	•	2	2	ş	9 3		2 <b>P</b> 0			Selent 710	ROM based OS Assembler editor, ticker far und en MD er SU eC boards	
	1100 8000	280 fant	8	2	-	-	2	2	ţ	1 3	1		Party	farluen	4.2	Haid dis haird Native der system Alsu for 50 boarts	

148

TABLE XIV (Continued)

	į	* 4 distemes * One inne assembler/	e-ustemu.er 8 Actos authmetic modula 1 8/6. 6/6 micromodula		- 8048-49-50. 8070 4 Multiprocesson 9070 184-1107 - 11 1911 184-1107 - 11 1911 184-1107 - 11 1911 184-1107 - 11 1911	Mag tape 5 Mb dish care.	Z margala channes for 7a products 22 foregraf 9 forefact tangang	Convertant Nucleone BOB Concepted E Law 218 10 Tured 258 computer	15 of 8 m. ('040'	15 20-145 455 • Start in 16 term net	At the server of	Proprietary Language	Percom 6809 EXORCIAN Adapter mout Crather Microtystems EXORCIAN	Percon SEC 9 Percon SEC 9 PEROPORE SWIP MER	Parcum U Diamay Munitory Auto-mus to second ROM VI Provision Fou-	restrict as days men				2 autories	fer treadul 2 MB	
	Į		Reinculable Macrosomer- bler & CRT Editor	_	Penertul editor, prompting menus BLCSBC bound laver computer support Stama (Stampter to Inter MUS stommunications lang)	Outlieital, full page test- rotter, 3270 comm. upput	Datalent, attach 9 CPUs	ruperior, unare dry 4 pe- rupherati- mag larg, USC SUC, async purities from 60 chrs to 600 kirsymm		fac antenes con to	system format hat edu		User-modif of commands: hardw , easily adaptable: faulture for for policiam dor: [[PROM: and for commands	INDEX and the mer are	intertuptis, user-espandable; supports serval ASCII Presitiocated, sandon accesa, contiguous file	her contants. 10 uses defined functions, comes w the net, day un	(JITE (BATK)		Extended memory, hended graphics	Macroassambler, mayn Morary (opt.)	Up to 256 territ. derigned for process con	Provined to support Cutics Revisives from
	Peripheral A	Cer -	Fronter Hrm.	CRT. TTY Stern 700	CAT (24 a D) A fictory dr. R5232C ter. PROM progr	CRT (1920 cher) 5 prunters.	2 CR1s (24 80) 2 comm	8 CR15 (24 #		CRT, herboard	line & char. printers.		CAT (A5.232), cesserie (AC), printer	VO-Indepen- dent <sup>5</sup>	Cassette, CMT Printer	LIKS 80 pe- ripherals		Printer 2. Nopper.	EPPOM burner commun. Kations	CAL, LTV Profes	CRT, printer	Locas NO
	Complian	Fortran	MPL. Bask Fortran		fortan	10 20 20 20 20 20 20 20 20 20 20 20 20 20		Cobul Basic			OF THE		VN	VNI .	VNV .	NA	Paxer	MA	N.A.	Bowell.		
Superior horse	Brilospedera	344	Parcel Parcel	Superlink (Linker)	99-1X	5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	A Provide Land	Cobel Basic			Page 1		N/N		11 1	i za za za	Fortran.	N.N	NA	Baue 1. 2. pforth	Process control Basic	
	Assemblers	2	89	-		VN	<b>V</b> N	MA.		Macro	Inhing. Relocating		<b>VN</b>	N.	84	¥1	997	NA	N/N	1802, 1804	N/A	- M
	4 :	2 MP.	- 4 ~	N.N	11	245 Mb. 4 ditte	1	SON NUC	1.2 Mb	160 Mu.	160 111.0	160 Mb	W	NA	MA	N/N	915 19	512 16	912 MP	02 <b>1</b>	a la la	A Mile
	(Tom)	2 3	2 3	9 3	1 1	256 W	4 3	41 542	23	3	3	43	1	6 tur	05 he	1	91 F3	91 J3	3	4	3	4
		2	ž	2	<u>}</u>	Yes	<b>10</b>	ž	ş	5	ş	Yes	Yes	×.	e,	ţ	4 F	2	<u>,</u>	2	\$	<u>ş</u>
	ij	£	£	2	2	*	¢ø),	8	Yes	r say	Yes.	, soy	ł	ž	1	2	Ş			2	£	°z
	Ĭ	ž	£	2	ž	- 3	e Server	2 1 1 1 1	£	ž	ž	2955	2	£	ł	ł	۴X	e,	ž	£	a Y	ş
1	5	-	-	-	•	•			~	-	-	-	-	-	-	-	-	-	-	-	-	-
1		-	-	-	-	2	*	2	-	-	-	3	-	-	-	-	-	-	-	-	-	
		•	•	2	•	-	•	1 1 1	11	ì	=	ű È	Z	2	2	۹.	æ 11	-	# 1 #	•	4 1 1	•
	3	83	6009	ş	Veed	(I-WI)		Spot	B060.	BOBU.	BUNG.	8080 1 A 2	8 3	3	8	92	8	8	3	2	0.4451	00%41
	Co Medica	EXORCISE EXORCISE	LXOR: No.	Microson Bule 19		20 moon	Madel 405	Madel 415	06-1/50/64/10	00-1-10-00	OP-1/44	OP 170 M.	-	•	ExtORcase	145-80	38008/2	A0066	V506E	RCA boards	14-00-510	5-046-11
	1 8	0089 500W	eges soom	Surfame	Starpter DOS	65-85	02 408 80	05 445	SOOM	SOOM	POMMOS	METOR	PSYMON	MDCX	ž	MICROCOS	2/90090	A006C	V-004	\$0 2081	1	Multi Critical
	5		100 M	1	Artend Ser	Mathins Systems	In State		Date Con.	Washing Mr			Pucon Pro	•					2		Print Med R.	(1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.

149

TABLE XIV (Continued)

ľ			ſ	ſ	Number of		-		-			Í	Massing any second			-	
		Les Monday	]		- Lun	-3	1	ż,	11			Assemblers	Interpreters	Compiliers	Peripherals	Comments	į
	3	00 00 00	060/200	V.N	•		ž	ž	2	12 10	-93 -1	B060. 280	MBasic	Bat Bave. Forth BO	6 CRTS GPID		18 drives
	SYSAUC	04 .06.708	082/0808	<b>N</b> N	-	-	ž	2	2	2	3-	B080, 280		Forth 80	LI CRT. I Ser chan. Girlb	Available with AMD9511 for fast math	
Sidla	MTS	09 09 09	987	< N	2	2	R Y	¥	5	4	1 66	Breo. 280	MBasic	MBasec Fortran 80. Cobol 80	16 CRTs. GPIB		
A 211 W	50 <b>3</b> 5	traces etco	05:00	ž	2	-	12	Future	ş	1 3	31	6800	ē	er. 10 dent BCD	CRT. Line printers. Diablo hytypu. dish drives	Interrupt druge UC. 	Winchester 10 Mb Carlinder dru. 10 Mb Compat Formers Multi terminal and book.
Antest Sys- Consultant 25/A. 1208 1 Ave. W	YN	Mb tarol a	6900	<b>u</b>	-		2	2	ş.	9	97 (T) 97	6809i	Basic. ml. Basic	¥¥	Terminel. disks	Also supports debug package. sort/merge. text formatting. fast interpreter	'Nallve and cross- atsembers *16 digit precision
PO Bon	1x999	1/066	1445 9900	-	-	~	ца,	ž	2	1	1 1 1 1 1 1 1	10066	Besc	Fortran		AMPL option for emulation and logic- state tracting	<sup>1</sup> Uker er pawlab'e <sup>2</sup> Avl Summer 1980 <sup>2</sup> 2-pars assembler
100(1	TIMMX	¥7066	1MS 9900	D F.H.R	~	~	ž	Xer,	¥.	4 3	1 MB (1450 P)	10066			CRT (24=80). CRT (12=80).	Multiprogrammung. math parkage	
	Beseco	5 990VA	1066 SM1	0 F.H.A	~	ñ	r Xa	Yes	\$ 1	91 35	0 5 M0 (2 disks)	99001 Basec	Basic		Line printer PROM burner. ANPL emutator	IF/THEN construct, math pack, direct access to bit level	
	Pescal	89004	TMS 990	H'L O	2	~	ž	λw.	ž	<del>1</del> 3	1 Mb (4 01345)	10066	Patcal	Patcal		Multypogrammag native code or mier- pretive: math pack, drect bit ihrut accase	

N/A: Not Applicable

(From: Electronic Design 24, Nov. 22, 1979)

·····

#### TABLE XV

#### Semiconductor Technologies

The purpose of this table is to present briefly the semiconductor technologies currently used in microprocessors.

#### a. Characteristics of a Semiconductor Technology

Speed 1. 5. Noise immunity 2. Density 6. Ruggedness 3. Cost 7. TTL compatibility 4. Power consumption Maturity or experience 8.

#### b. Comparison of Semiconductor Technologies

Tech- nology	Speed 1 = fast	Density l = complex	Cost 1 = cheap	Power 1 = lower	Ruggedness 1 = most	Experience 1 = long	TTL Compat- ibility
CMOS	3	3	3	1	1	4	Yes
ECL	1	6	6	6	6	5	No
TTL Scho.	. 2	5	3	5	2	1	Yes
I <sup>2</sup> L	3	3	3	2	3	6	?
NMOS	5	1	2	3	4	3	Sometimes

# c. Application Requirements and Appropriate µP Technologies

Requirement	Most Suitable Technologies
Low cost	PMOS, NMOS
Small size	PMOS, NMOS
High speed	ECL, TTL Schottky
Low power consumption	CMOS
Rugged environments	CMOS
Compatibility with: TTL CMOS ECL	TTL Schottky, CMOS TTL Schottky, CMOS ECL
Wide availability	PMOS, NMOS
Standard parts in same technology	TTL Schottky, CMOS, ECL
Large memories in same technology	PMOS, NMOS, TTL Schottky
Most support	PMOS, NMOS

151

# TABLE XV (Continued)

ar onergoeo.	200200			10105103
Typical Gate Size (sq.mills)*	Typical Delay/Gate (nanosec.)	Typical Power Gate	Typical Cost/Gate (cents)	Typical Noise Immunity (volts)
3	100 ns	0.2 mW	0.1-2	1.0
2	50 ns	0.2 mW	0.1-2	0.4
12	25 ns	10 µW	10-30	4.0
13	10 ns	10 mW	5-15	0.4
TL 5	3 ns	20 mW	25	0.3
5 TL	10 ns	2 mW	25	0.3
8	2 ns	30 mW	30-40	0.125
1	25 ns	50 µW	5-50	0.2
	Typical Gate Size (sq.mills)* 3 2 12 13 TL 5 5 TTL 8 1	Typical Gate Size (sq.mills)*     Typical Delay/Gate (nanosec.)       3     100 ns       2     50 ns       12     25 ns       13     10 ns       5     10 ns       7     10 ns       12     25 ns       13     10 ns       5     10 ns       7     3 ns       10 ns     10 ns       1     25 ns	Typical Gate Size (sq.mills)*     Typical Delay/Gate (nanosec.)     Typical Power Gate       3     100 ns     0.2 mW       2     50 ns     0.2 mW       12     25 ns     10 µW       13     10 ns     10 mW       FL     5     3 ns     20 mW       5     10 ns     2 mW       TTL     8     2 ns     30 mW       1     25 ns     50 µW	Typical Gate Size (sq.mills)*     Typical (nanosec.)     Typical Power Gate     Typical Cost/Gate (cents)       3     100 ns     0.2 mW     0.1-2       2     50 ns     0.2 mW     0.1-2       12     25 ns     10 µW     10-30       13     10 ns     10 mW     5-15       FL     5     3 ns     20 mW     25       5     10 ns     2 mW     25       TTL     8     2 ns     30 mW     30-40       1     25 ns     50 µW     5-50

#### d. Typical Characteristics of Semiconductor Technologies

\* One square mil. =  $6.45 \times 10^{-6}$  square centimeters.

# e. Types of Semiconductor Memory

Memory Type	Variability	Technologies	Volatility	Typical Size (bits)	Use
ROM	Fixed	A11	Nonvolatile	16K	Program memory, tables
PROM	Programmed once	Most	Nonvolatile	1K to 16K	Program memory, tables
EPROM	Can be reprogramme	MOS ed	Nonvolatile	2K to 32K	Program memory, tables
EAROM	Can be reprogramme	Few ed	Nonvolatile	1K	Program memory, tables
RAM	Variable	A11	Volatile	1K to 16K	Data
Shift register	Variable	A11	Volatile	8-256	Input/output
Buffer	Variable	Most	Volatile	32-256	Input/output

f. Available Semiconductor RAMs

Technology	Maximum Size (bits)	Typical Size (bits)	Typical Access Time	Dynamic or Static	Typical Cost
PMOS	4K	1K	1 µs	Both	\$3/1K bits
NMOS	16K	1K	300-500 ns	Both	2/1K bits
CMOS	1K	256	100-500 ns	Static	20/1K bits
Schottky TT	L 1K	256	50-100 ns	Static	20/1K bits
ECL	1K	128	10-50 ns	Static	15/128 bits
I <sup>2</sup> L	4K	-	100 ns	Static	-

152

- Sittal Hol

# TABLE XV (Continued)

Technology	Maximum Size (bits)	Typical Size (bits)	Typical Access Time	Typical Cost
PMOS	64K	16K	800 ns 1.5µs	\$25/16K
NMOS	64K	16K	100 ns 1 µs	25/16K
CMOS	12K	1 K	100-500 ns	50/12K
Schottky TTI	16K	1K	50-100 ns	10/1K

# g. Available Semiconductor ROMs

# h. Available PROMs

Technology	Sizes (bits)	Typical Access Times	Typical Programming Time	Typical Cost
PMOS	1.K-8K	1-2 µs	2 min	\$20/2K
NMOS	1K-32K	300 ns – 1 µs	2 min	20/8K
TTL	512-4K	50 - 100 ns	30 s	22/1K
Schottky TTL	512-16K	50 - 100 ns	30 s	6/1K
ECL	1K	10 - 50 ns		

# i. <u>Typical Shift Registers</u>

Technology	Size	es_	(bits)	Maximum Operating Frequency (MHZ)	Cost
PMOS	up	to	2 K	1	\$5/1K
NMOS	up	to	2 K	1-10	\$5/single 1K
CMOS	up	to	200	5	\$6/200 bits
Schottky TTL	. 4	or	8	50-70	\$2-\$10
ECL	4	or	8	100	\$10-\$15

TABLE XVI



Worksheet for a first-pass evaluation of  $\mu P/\mu C$ -chip families. The attempt at such an evaluation-reflected in the rankings within each application area-hinges on the criterion that high-volume production makes a  $\mu P$  or  $\mu C$  a safe bet; technical appeal is the secondary criterion. Lines connecting chips in the different application areas show which families have the broadest application flexibility (solid link with relatively strong family relationship and dashed link chips with weaker ties.)

(From: EDN, October 20, 1979)

154

HOLE AND



Production volumes for many  $\mu$ Ps and  $\mu$ Cs soared in 1979. The older PMOS 1-chip  $\mu$ Cs continue to lead in volume, with the TMS 1000 remaining ahead of all competition by a wide margin. In the 8-bit 1-chip- $\mu$ C field, Intel's 8048 is coming on strong and could surpass the 8080 in overall profitability. In the 16-bit- $\mu$ P field, trends haven't developed with regard to the 8086, 28000 and MC68000. (From: EDN, Oct. 20, 1979).

155

in California and

### TABLE XVIII

# LIST OF ORIGINAL-SOURCE µP/µC MANUFACTURERS

Name	Address	Telephone
Advanced Micro Devices	901 Thompson Pl Sunnyvale, CA 94086	(408)732-2400
AEG Telefunken	6 Frankfurt 70 AEG Hochhaus, Federal Republic of Germany	
American Micro-Systems Inc.	3800 Homestead Rd, Santa Clara, CA 95051	(408)246-0330
Analog Devices, Inc.	P.O. Box 208, Norwood, MA 02062	(617)329-4700
Burr-Brown	P.O. Box 11400, Tucson, AZ 85734	(602)294-1431
Data General Corp.	9 Route, Westboro, MA 01581	(617)366-8911
EMM Semiconductor	3883 N. 28th Ave. Phoenix, AZ 85017	(602)263-0202
ESSEX International	301 Alpha Drive Pittsburgh, PA 15328	(412)963-9322
Fairchild Semi- conductor	464 Ellis St. Mountain View, CA 94040	(415)962-3541
Ferranti Electric	E. Bethpage Road Plainview, NY 11803	(516)293-8383
Ferranti Ltd.	Western Rd., Bracknell Berkshire RG121RA England	•••
Fujitsu America, Inc.	1208 E. Arques Ave. Sunnyvale, CA 94086	(408)739-3200
Fujitsu, Ltd.	6-l Marunochi 2 Chome Chiyoda-Ku, Tokyo, Japan	

ł

156

-

Name	Address	Telephone
General Instrument Corp.	600 W. John Street Hicksville, NY 11802	(516)733-3130
Harris Semiconductor	P.O. Box 883 Melbourne, FL 32901	(305)727-5407
Hitachi America Ltd	707 W. Algonquin Rd. Arlington Heights Il 60005	(312)593-7660
Hitachi Ltd	Nippon Bldg. No. 602 2-Chome, Ohtemachie, Chiyoda-Ku, Tokyo 100 Japan	••••
Hughes, Solid State Division	2601 Campus Drive Irvine, CA 92715	(714)752-6396
Intel Corp.	3065 Bowers Ave. Santa Clara, CA 95051	(408)987-8080
Intersil Inc.	10900 N. Tantau Ave. Cupertino, CA 95014	(408)996-5000
ITT Semiconductor	74 Commerce Way Woburn, MA 01801	(617)935-7910
ITT Semiconductor	Maidstone Road Footscray, Sidcup Kent, England	
Matrox Electronic Systems Ltd	2795 Bates Rd. Montreal, Quebec H35-1B5, Canada	(514)481-6838
Micro Networks Corp.	324 Clark St. Worcester, MA 01606	(617)852-5400
Mitel Semiconductor	18 Airport Blvd. Bromont, Quebec, JOE-1LO, Canada	<u>(</u> 514) 534- 2321
Mitel Semiconductor	2321 Morena Blvd. Suite M, San Diego CA 42110	(714)276-3421
Monolithic Memories, Inc.	1165 E. Arques Ave. Sunnyvale, CA 94086	<u>(</u> 408)739-3535

Name	Address	Telephone
MOS Technology, Inc.	950 Rittenhouse Rd. Norristown, PA 19401	(215)666-7950
Mostek Corp.	1215 W. Crosby Rd. Currollton, TX 75006	(214)242-0444
Motorola Semiconductor	5005 E. McDowell Rd. Phoenix, AZ 85008	(602)244-6900
Motorola Integrated Circuits	3501 Ed Bluestein Blvd. Austin, TX 78721	(512)928-6800
National Semiconductor Corp.	2900 Semiconductor Dr. Santa Clara, CA 95051	(408)737-5000
NEC Microcomputers, Inc.	173 Worcester St. Wellesley, MA 02181	(617)237-1910
Panafacom Ltd.	2-10-16 Jiyuzaoka Mezuro-Ku, Tokyo, Japan	
Panasonic	50 Meadowland Parkway Secancus, NJ 07094	(201) 348-7276
Philips Industries, Electronic Comp. and Materials Div.	P.O. Box 523, Eindhoven The Netherlands	
Raytheon Semiconductor Div.	350 Ellis St., Mountain View, CA 94042	(415)968-9211
RCA Solid State Div.	P.O. Box 3200, Rte.202 Somerville, NJ 08876	(201)685-6423
Rockwell International	P.O. Box 3669, RC01 Dept. 720, Anaheim, CA 92803	(714)632-2321
Scientific Micro Systems	520 Clyde Ave., Mt. View, CA 94043	(415)964-5700
SGS-ATES	Via C Olivetti 1/20041 Agrate Brianza, Italy	
Sharp	22-22 Nagaike-Cho, Abeno-Ku, Osaka 545 Japan	
Sharp Electronic	10 Keystone Place Paramus, NJ 07652	(201)265-5600

winding the

Name	Address	Telephone
Siemens AG. Central Information Department	Oskar-Von-Miller Ring 18, D-8000, Munich 2, Federal Republic of Germany	
Siemens Corp.	186 Wood Ave., South Iselin, NJ 08830	(201)494-1000
Signetics	811 E. Arques Ave. Sunnyvale, CA 94086	(408)739-7700
Solid State Scientific Inc.	Montgomeryville Industrial Park Montgomeryville, PA 18936	(215)855-8400
Standard Microsystems	35 Marcus Blvd. Hauppauge, NY 11787	(516)273-3100
Synertek	3050 Coronado Drive Santa Clara, CA 95051	(408)241-4300
Texas Instruments	13500 North Central Expressway, M/S 308 Dallas, TX 75222	(214)238-2011
Thomson-CSF, Sescosem	101 Boulevard Murat 75781 Paris Cedex 16 France	
Toshiba America, Inc.	2900 McArthur Blvd. Northbrook, IL 60062	(312)564-5140
Toshiba Transistor Works	l Komukai, Toshiba-cho Kawasaki-shi, Kanaganaken, Japan	
TRW LSI Products	P.O. Box 1125, Redondo Beach, CA 90278	(213)535-1831
Western Digital Corp.	3128 Red Hill Ave. Newport Beach, CA 92663	<u>(</u> 714) 557-3550
Zilog Microcomputers	10460 Bubb Rd. Cupertino, CA 95014	(408)446-4666

#### GLOSSARY

Access Time - The delay between the time when a memory receives an address and the time when the data from that address is available at the outputs.

Accumulator - A register that is the source of one operand and the destination of the result for most arithmetic and logical operations.

Active-High - The active state is the one state.

Active-Low - The active state is the zero state.

- Ada High order level language designed for the necessities of DOD.
- Address The identification code that distinguishes one memory location or input/output port from another and that can be used to select a specific one.
- Addressing Methods (Modes) The methods for specifying the addresses to be used in an instruction. Common addressing methods include direct, indirect, indexed, relative, and stack.
- ALGOL Algorithmic Language, a widely used high-level language designed for systems and scientific applications.
- Analog Continuous signal or representation of a quantity that can take any value.

Anode - Positive terminal.

- Architecture Structure of a system. Computer architecture often refers specifically to the CPU.
- Arithmetic-Logic Unit (ALU) A device that can perform any of a variety of arithmetic or logical functions under the control of function inputs.
- Arithmetic Shift A shift operation that preserves the value of the sign bit (most significant bit).
- Arm See Enable, but particularly applied to enabling interrupts.
- ASCII American Standard Code for Information Interchange, a 7-bit character code widely used in computers and communications.

- Assembler A computer program that converts assembly language programs into a form (machine language) that the computer can understand. The assembler translates mnemonic instruction codes into binary numbers, replaces names with their binary equivalents, and assigns locations in memory to data and instructions.
- Assembly Language A programming language in which the programmer can use mnemonic instruction codes, labels, and names to refer directly to their binary equivalents. The assembler is a low-level language, since each assembly language instruction translates directly into a specific machine language instruction.
- Asynchronous Operating without reference to an overall timing source, that is, operating at irregular intervals.
- Attached Input/Output An addressing method for input/output ports that identifies the ports either directly (if the port is attached to the CPU) or from the address in memory to which the port is attached. The port is usually selected with special instructions that are decoded either in the CPU or in the memory section. Systems using attached I/O are frequently based on LSI devices that combine memory, input/output, and processor functions.
- Auto-Correlation Functions (ACF) Output of the transmitted signal.
- Autoindex An index register that is automatically incremented or decremented with each use.
- Auxiliary Carry Bit See Half-Carry Bit.
- Bank A directly addressable set of registers or memory locations. The register or other storage device that selects banks is called a "bank switch."
- Baud A communications measure for serial data transmission, bits per second but including both data bits and bits used for synchronization, error checking and other purposes.
- Baud Rate Generator A device that generates the proper timing interval between bits for serial data transmission.
- Baudot Code A 5-bit character code used in telegraphy and some communications terminals.
- BCD (Binary-Coded Decimal) A method for representing decimal numbers whereby each decimal digit is separately coded into a binary number.

Benchmark Program - A sample program used to evaluate and compare computers.

Bidirectional - Capable of transporting signals in either direction.

Binary - A binary digit, possible values zero or one.

- Bit Manipulation (or Bit Banging) The examination and changing of single bits or small groups of bits within a word.
- Bit Slice A section of a CPU that may be combined in parallel with other such sections to form complete CPUs with various word lengths.
- Bootstrap Loader (or Bootstrap) Technique for loading first instructions of a program into memory and then using these instructions to bring in the rest of the program. The first instructions (called the "bootstrap") may reside in a special read-only memory.
- Borrow A status bit that is one if the result of an unsigned subtraction was negative.
- Bottom-Up Design A design method in which parts (or modules) of a system are designed and tested separately before being combined.
- Bounce Moving back and forth between states before reaching a final state.

Branch Instruction - See Jump Instruction.

- Breakpoint A location specified by the user at which program execution is to end temporarily. Used as an aid in program debugging.
- Bus A group of parallel lines that connect two or more devices.
- Bus Contention A situation in which two or more devices are trying to place data on a bus at the same time.
- Bus Driver A device that amplifies outputs sufficiently so that they can be recognized by the devices on a bus.
- Bus Isolation Buffering parts of the bus away from other parts with buffers and drivers.
- Bus Transceiver A device that acts as both a bus driver and bus receiver; that is, it interfaces a bidirectional bus to two unidirectional buses.

Byte - The basic grouping of bits that the computer handles as a unit, most often eight bits in length.

Call - See Subroutine.

- Carry Bit A status bit that is one if the last operation generated a carry from the most significant bit.
- Cartridge (or 3M Mag-Tape Cartridge) A compact, enclosed package of magnetic tape that uses 1/4-inch tape and records 1600 bits per inch at 30 in./s on four tracks.
- Cassette An enclosed package of magnetic tape usually housed in a plastic container. Both audio and digital versions exist; the digital ones are more reliable and more expensive. The standard unit is the Philips-type cartridge, which consists of 282 feet of 0.015-in. magnetic tape, phase encoded at 800 bits per inch.

Cathode - Negative terminal.

- Central Processing Unit (CPU) The control section of a computer. It contains the arithmetic unit, registers, instruction-decoding mechanism, and timing and control circuitry.
- Checksum A logical sum of data that is included in a record as a guard against recording or transmission errors.
- Chip A substrate containing a single integrated circuit.
- Clear Set state to zero; an input to a device that sets the state to zero.
- Clock A regular timing signal that governs transitions in a system.
- CMOS Complementary metal-oxide semiconductor, a logic family that uses complementary N-channel and P-channel MOS field-effect transistors to provide high noise immunity and low power consumption.
- Coding The writing of programs in a language that is comprehensible to a computer system.
- Common-Anode Display A multiple display in which signals are applied to the cathodes of the individual displays and the anodes are tied together to the power supply; uses negative logic (i.e., a logic zero turns a display on).
- Common-Cathode Display A multiple display in which signals are applied to the anodes of the individual displays and the cathodes are tied together to ground. Uses positive logic (i.e., a logic one turns a display on).

. Standar

Common I/O - Uses the same lines for input and output.

- Communications Register Unit (CRU) The general-purpose command-driven hardware interface of TI's 990/9900  $\mu P$  family.
- Comparator A device that produces outputs that show whether one input is greater than, equal to, or less than the other input. Both analog and digital comparators exist.
- Compiler A program that converts a program in a high-level or procedure-oriented language into an assembly or machine language program.

Computer-aided design (CAD).

- Concurrent characteristics One of several characteristics associated with the definition of a process in  $\mu P$  PASCAL, including the process priority, the stack size and the heap size.
- Condition Code (or Flag) A single bit that indicates a condition within the computer, often used to choose between alternate instruction sequences.
- Condition Code Register A register that contains one or more condition codes.
- Control Memory A memory that holds microprograms--that is, a memory used to decode computer instructions.
- Core Memory A magnetic memory that can be magnetized in one of two directions so as to represent a bit of data.
- Counter A clocked device that enters a different state after each clock pulse (up to its capacity) and produces an output that reflects the total number of clock pulses it has received. Counters are also referred to as "dividers," since they divide the input frequency by "n" where "n" is the capacity of the counter.
- Cross-Assembler An assembler that runs on a computer other than the one for which it assembles programs.
- CRT Cathode-ray tube.
- Current Page The page of memory on which the present instruction is located.
- Current-Loop Interface (or Teletype Interface) An interface that allows connections between digital logic and a device that uses current-loop signals--that is, typically the presence of 20 mA in the loop is a logic one and the absence of that current is a logic zero.

12 and to in the

Cycle Stealing - Using a cycle during which the CPU is not accessing the memory for a DMA operation.

- Cycle Time Time interval at which a set of operations is repeated regularly in the same sequence.
- Cyclic Redundancy Check (CRC) An error-detecting code generated from a polynomial that can be added to a data record or sector.
- Daisy-Chain An input or output method whereby signals pass from one device to another until accepted or blocked. Activity near the control unit for the chain will block activity farther from the control unit.
- Data Acquisition System A system that will accept several analog inputs and produce corresponding digital data. The system usually includes sample and hold circuitry, multiplexers, and converters.
- Data Fetch Cycle A computer operation cycle during which data is brought from memory to the CPU.
- Data Pointer (or Pointer) A register or memory location that holds an address rather than the data used.
- Debounce Convert the output from a contract with bounce into a single, clean transition between states.
- Debounce Time The amount of time required to debounce a closure.
- Debug To eliminate programming errors, sometimes referred to as verifying the program.
- Debugger (or Debug Program) A program that helps in finding and correcting errors in a user program.
- Decade Counter A counter with ten different states.
- Decimal Adjust An operation that converts a binary arithmetic result to a decimal (BCD) result.
- Decoder A device that produces unencoded outputs from coded inputs.
- Delay Time The amount of time between the clocking signal and the actual appearance of output data, or the time between input and output.
- Demultiplexer A device that directs a time-shared input to one of several possible outputs, according to the state of the select inputs.

DOD - Department of Defense.

- Destructive Readout (DRO) The contents cannot be determined without changing them.
- Development System A special computer system that includes hardware and software specifically designed for developing programs and interfaces.
- Device, Logical An entity with which programs can perform device-independent I/O.
- Device, Physical An entity that communicates with programs via a CRU or memory-mapped I/O and interrupts.
- Diagnostic (Program) A program that checks part of a system for proper operation.
- Digital Having discrete levels, quantized into a series of distinct levels.
- Direct Addressing An addressing method whereby the address of the operand is part of the instruction.
- Directly Addressable Can be addressed without changing the contents of any registers or bank switches.
- Direct Execution A method whereby the computer directly executes statements in a high-level language rather than translating those statements into machine or assembly language.
- Direct Memory Access (DMA) An input/output method whereby an external controller directly transfers data between the memory and input/output sections without processor intervention.
- Disable Prohibit an activity from proceeding or a device from producing data outputs.
- Disable Time (Output) The amount of time required for an active tri-state output to enter the third or opencircuit state.
- Disarm See Disable, but particularly applied to disabling interrupts.
- DFT Discrete Fourier Transforms.
- Disk Operating System (DOS) An operating system that transfers programs and data to and from a disk, which may be either flexible or fixed-head; the operating system may itself be largely resident on the disk.

+ ... Reader

Diskette - See Floppy Disk.

Divider - See Counter.

Dual Inline Package (DIP or Bug) - A semiconductor chip package having two rows of pins perpendicular to the edges of the package, sometimes called a "bug," since it appears to have legs.

- Dynamic Memory A memory that loses its contents gradually without any external causes.
- EAROM Electrically alterable ROM, a nonvolatile RAM, often with a relatively long write time.
- EBCDIC Expanded Binary-Coded Decimal Interchange Code, an 8-bit character code often used in large computers.
- ECL Emitter-coupled logic, a high-speed bipolar technology often used in computer mainframes.
- Editor A program that manipulates text material and allows the user to make corrections, additions, deletions, and other changes.
- Effective Address The actual address used by a particular instruction to fetch or store data.
- Emulator A microprogrammed copy of an existing system.
- Enable Allow an activity to proceed or a device to produce data outputs.
- Encoder A device that produces coded outputs from unencoded inputs.
- EPROM (or EROM) Erasable PROM, a PROM that can be completely erased by exposure to ultraviolet light.
- Error-Correcting Code A code that can be used by the receiver to correct errors in the messages to which the code is attached; the code itself does not contain any additional message.
- False Start Bit A start bit that does not last the minimum required amount of time, usually caused by noise on the transmission line.

Fan-In - The number of inputs connected to a gate.

Fan-Out - The maximum number of outputs of the same family that can be connected to a gate without causing current overload.

FFAT - Fast Fourier Transforms.

- Field-Programmable Logic Array (FPLA) A programmable logic array that can be programmed by the user.
- FIR Finite Impulse Response, filters.
- Firmware Microprograms, usually implemented in read-only memories.
- Fixed-Instruction Computer A computer for which the manufacturer determines the instruction set. As opposed to microprogrammable computer.

Fixed Memory - See ROM.

- Flag See Condition Code.
- Flatpack A semiconductor chip package in which the pins are in the same plane as the package rather than perpendicular to it as in a DIP.
- Flip-Flop A digital electronic device with two stable states that can be made to switch from one state to the other in a reproducible manner.
- Floating Not tied to any logic level, often applied to tri-state outputs that are in the high-impedance state. TTL devices usually interpret a floating input as a logic one.
- Floppy Disk (or Flexible Disk) A flexible magnetic surface that can be used as a data storage device; the surface is divided into sectors. An IBM-compatible floppy disk is one that uses formatting and sectoring techniques originally introduced by IBM. The individual floppy disk is sometimes called a "diskette."
- Flowchart A graphical representation of a procedure or computer program.
- FORTRAN A high-level (procedure-oriented) programming language devised for expressing scientific problems in algebraic notation. Short for Formula Translation Language.
- FTR Functional Throughout Rate.
- Gate A digital logic element where the binary value of the output depends on the values of the inputs according to some logic rule.

GaAs - Gallium-Arsenide gates.

General-Purpose Interface Bus (GPIB or Hewlett-Packard Bus) -A standard interface for the transmission of parallel data in a network of instruments. The GPIB has 8 data lines, 8 control lines, and 8 ground lines.

General-Purpose Register - A register that can be used for temporary data storage.

Gray Code - A binary code sequence in which only one bit changes in a transition to the next higher or lower value.

Half-Carry (or Auxiliary Carry) Bit - A status bit that is one if the last operation produced a carry from bit 3 of an 8-bit word. Used on 8-bit microprocessors to make the correction between binary and decimal (BCD) arithmetic.

Hardware - Physical equipment forming a computer system.

Heap - A data area holding dynamically allocated variables.

Hex - (1) Containing six distinct logic elements, as in hex buffers; (2) abbreviation for hexadecimal or base 16.

Hexadecimal - Number system with base 16. The digits are the decimal numbers 0 through 9, followed by the letters A through F.

High-Impedance State - See Tri-State.

HLL (High-Level Language, or Procedure-Oriented Language) -A programming language for which the statements represent procedures rather than single machine instructions. FORTRAN, COBOL, and BASIC are three common high-level languages. A high-level language requires a compiler that translates each statement into a series of machine language instructions.

HOL - High Order Language (same as HLL).

Hold Time - The amount of time after the end of an activity signal during which some other signal must be stable to ensure the achievement of the correct final state.

IEEE Standard 488 Bus - See General-Purpose Interface Bus.

Immediate Addressing - An addressing method in which the operand is part of the instruction itself.

Immediate Data - Data that is part of the instruction that uses it.

169

Implied (or Inherent) Addressing - The operation code itself specifies all the required addresses.

- In-Circuit Emulator A device that allows a prototype to be attached to a development system for testing and debugging purposes.
- Index Register A register that can be used to modify memory addresses.
- Indexed Addressing An addressing method in which the address included in the instruction is modified by the contents of an index register in order to find the actual address of the data.
- Indirect Addressing An addressing method in which the address of the data, rather than the data itself, is in the memory location specified by the instruction.
- Input/Output (Section) The section of the computer that handles communications with external devices.
- Instruction A group of bits that defines a computer operation and is part of the instruction set.
- Instruction Cycle The process of fetching, decoding, and executing an instruction.
- Instruction Execution The process of performing the operations indicated by an instruction.
- Instruction (Execution) Time The time required to fetch, decode, and execute an instruction.
- Instruction Fetch The process of addressing memory and reading an instruction word into the CPU for decoding.
- Instruction Length The number of words of memory needed to store a complete instruction.

Instruction Repertoire - See Instruction Set.

- Instruction Set The set of general-purpose instructions available with a given computer--that is, the set of inputs to which the CPU will produce a known response during the instruction fetch cycle.
- Integrated Circuit (IC) A complete circuit on a single substrate or chip.
- I<sup>2</sup>L Integrated-injection logic, a bipolar technology that uses only transistors (both vertical and lateral) to provide moderate speed, low power consumption, and high density.

- Intelligent Terminal (or Smart Terminal) A terminal that has some data processing capability or local computing capability.
- Interpreter A program that fetches and executes instructions written in a high-level language. An interpreter executes each instruction as soon as it reads the instruction; it does not produce an object program, as a compiler does.
- Interrupt A computer input that temporarily suspends the normal sequence of operations and transfers control to a special routine.
- Interrupt-Driven System A system that depends on interrupts to handle input and output or that idles until it receives an interrupt.
- Interrupt Mask (Interrupt Enable) A mechanism that allows the program to specify whether interrupts will be accepted.
- Interrupt Service Routine A program that performs the actions required to respond to an interrupt.

Inverter - A logic device that complements the input.

- Isolated Input/Output An addressing method for I/O ports that uses an addressing system distinct from that used by the memory section.
- Jump Instruction An instruction that places a new value in the program counter, thus departing from the normal onestep incrementing. Jump instructions may be conditional; that is, the new value may only be placed in the program counter if certain conditions are met.
- Jump Table A table that contains the addresses of routines to which the computer can transfer control.
- $K 2^{10}$  or 1024 words, a unit of memory.

Keyboard - A collection of key switches.

- Keyboard Encoder A device that produces a unique output code for each possible closure on a keyboard.
- Keyboard Scan The process of examining the rows and columns of a matrix keyboard to determine which keys have been pressed.

Kilobit - 1000 bits.

Label - A name attached to a particular instruction or statement in a program that identifies the location in memory of the object code or assignment produced from that instruction or statement.

THE REAL PROPERTY IN

Large-Scale Integration (LSI) - An integrated circuit with complexity equivalent to over 100 ordinary gates.

- Latch A temporary storage device controlled by a timing signal. The contents of the latch are fixed at their current values by a transition of the timing signal (clock) and remain fixed until the next transition.
- Light-Emitting Diode (LED) A semiconductor device that emits light when biased in the forward direction.
- Linear Select Using coded bus lines individually for selection purposes rather than decoding the lines. Linear select requires no decoders but allows only "n" separate devices to be connected rather than 2<sup>n</sup>, where n is the number of lines.
- Linking Loader A loader that will enter a series of programs and subroutines into memory and provide the required interconnections.
- Loader A program that reads a user or system program from an input device into memory.
- Logic Analyzer A test instrument that detects and displays the state of parallel digital signals.
- Logic Design Design using digital logic circuits.
- Logic Shift A shift operation that places zeros in the empty bits.
- Logical Sum A bit-by-bit EXCLUSIVE-ORing of two binary numbers.
- Lookahead Carry A device that forms the carry bit from a binary addition without using the carries from each bit position.
- Loop A self-contained sequence of instructions that the processor repeats until a terminal condition is reached. A conditional jump instruction can determine if the loop should be continued or terminated.
- Low-Level Language A language in which each statement is directly translated into a single machine language instruction. See Assembly Language and Machine Language.

Low-Power Schottky TTL - A low-power variant of standard TTL. Machine Code - See Machine Language.

Machine Cycle - The bacic (DU cycle - One rach

Machine Cycle - The basic CPU cycle. One machine cycle is the time required to fetch data from memory or execute a single-word operation.

- Machine Language The programming language that the computer can directly understand with no translation other than numeric conversions. A machine language program can be loaded into memory and executed. The value of every bit in every instruction in the program must be specified.
- Macro A name that represents a sequence of instructions. The assembler replaces a reference to the macro with a copy of the sequence.

Macroassembler - An assembler that has facilities for macros.

Macroinstruction - An overall computer instruction fetched from the main memory in a microprogrammed computer.

MBM - Magnetic Bubble Memory.

Majority Logic - A combinational logic function that is true when more than half the inputs are true.

Mark - The one state on a serial data communications line.

Mask - (1) A glass photographic plate that defines the diffusion patterns in integrated circuit production. (2) A bit pattern that isolates one or more bits from a group of bits.

Maskable Interrupt - An interrupt that the system can disable.

Matrix Keyboard - A keyboard in which the keys are connected in rows and columns.

MTBF - Mean time between failures.

MTTR - Mean time to repair.

Medium-Scale Integration (MSI) ~ An integrated circuit with a complexity of between 10 and 100 gates.

Megabit - One million bits.

- Memory (Section) The section of a computer that serves as storage for data and instructions. Each item in the memory has a unique address that the CPU can use to fetch it.
- Memory Address Register (or Storage Address Register) A register that holds the address of the memory location being accessed.
- Memory-Mapped Input/Output An addressing method for I/O ports that uses the same addressing system as that used by the memory section.
- Meta-Assembler An assembler for which the input instruction patterns can be defined and that can, therefore, assemble programs for different computers.

- Microassembler An assembler specifically designed for writing microprograms.
- Microcomputer A computer whose CPU is a microprocessor. A microprocessor plus memory and input/output circuitry.
- Microcontroller A microprogrammed control system without arithmetic capabilities.
- Microinstruction One of the words in a control memory-that is, one of the organized sequence of control signals that form the instructions at the control level.
- Microprocessor The central processing unit of a small computer, implemented on one or a few LSI chips.
- Microprocessor Analyzer A piece of test equipment that can be used to trace and debug the operations of a microprocessor.
- Microprogram A program written at the control level and stored in a control memory.
- Microprogrammable Having a microprogrammed control function that the user can change. That is, the user can add, enter, or replace microprograms.
- Microprogrammed Having the control function implemented through microprogramming.
- Microprogramming The implementation of the control function of a processing system as a sequence of control signals that is organized into words and stored in a control memory.
- Mnemonics Symbolic names or abbreviations for instructions, registers, memory locations, etc., which suggest their actual functions or purposes.
- Modem Modulator/demodulator, a device that adds or removes a carrier frequency, thereby allowing data to be transmitted on a high-frequency channel or received from such a channel.
- Modular Programming A programming method whereby the entire task is divided into logically separate sections or modules.
- Monitor A simple operating system that allows the user to enter or change programs and data, to run programs, and to observe the status of the various sections of the computer.

P. L. Marker All Ar

Monostable Multivibrator (or One-Shot) - A device that produces a single pulse of known length in response to a pulse input.

- MOS Metal-oxide semiconductor, a semiconductor process that uses field-effect transistors in which the current is controlled by the electric field around a gate.
- Multiplexer (or Selector) A device that selects one of several possible inputs to be placed on a time-shared output bus according to the state of the select inputs.
- Multiprocessing Utilizing two or more processors in a single system, operating out of a common memory.

MAC - Multiplier-accumulator.

Nanosecond -  $10^{-9}$  second, abbreviated ns.

- Negative Logic Circuitry in which a logic zero is the active or ON state.
- Nesting Constructing subroutines or interrupt service routines so that one transfers control to another and so on. The nesting level is the number of transfers required to reach a particular routine without returning.

Nibble - A sequence of four bits operated on as a unit.

- N-Key Rollover (NKRO) Resolving any number of simultaneous key closures into consecutive output codes.
- NMOS N-channel metal-oxide semiconductor, a logic family that uses N-channel MOS field-effect transistors to provide high density and medium speed.
- Noise Margin The noise voltage required to make logic circuits malfunction.
- Nondestructive Readout (NDRO) The contents of the device can be determined without changing those contents.
- Nonmaskable Interrupt An interrupt that the system cannot disable.
- Nonvolitile Memory A memory that does not lose its contents when power is removed.
- No-Op (or No Operation) An instruction that does nothing other than increment the program counter.

----
- Object Program (or Object Code) The program that is the output of a translator program, such as an assembler or compiler. Usually a machine language program ready for execution.
- Octal Number system with base 8. The digits are the decimal numbers 0 through 7.
- Offset A number that is to be added to another number to calculate an effective address.
- One-Address Instruction An instruction in which only one data address must be specified. The other data, if necessary, is presumed to be in the accumulator.
- One's Complement A bit-by-bit logical complement of a binary number.

One-Shot - See Monostable Multivibrator.

- On-Line System A computer system in which information reflecting current activity is introduced as soon as it occurs.
- Open-Collector Output A special output that is active-low but not high. Such outputs can be wire-ORed to form a bus employing negative logic.
- Operating System System software that controls the overall operation of a computer system and performs such tasks as memory allocation, input and output distribution, interrupt processing, and job scheduling.
- Operation Code (Op Code) The part of an instruction that specifies the operation to be performed during the next cycle.
- Optoisolator Semiconductor device consisting of an LED and a photodiode or phototransistor in close proximity. Current through the LED causes internal light emission that forces current to flow in the phototransistor. Voltage differences have no effect because the devices are electrically separated.
- Overflow Bit A status bit that is one if the last operation produced a two's complement overflow.
- Overlay The section of a program that is actually resident in memory at a particular time. A large program can be divided into overlays and run on a computer having limited memory but backup storage for the rest of the program.

Page - A subdivision of the memory section.

Page Zero - The first page of memory; the most significant address bits (or page number) are zero.

Parallel - More than one bit at a time.

- Parity A 1-bit code that makes the total number of one bits in the word, including the parity bit, odd (odd parity) or even (even parity).
- Parity Bit A status bit that is one if the last operation produced a result with even (if even parity) or odd (if odd parity) parity.

Passing Parameters - See Subroutine.

- Pipelining Overlapping cycles so that different parts of consecutive cycles are performed at the same time.
- PL/I Programming Language 1, a high-level language developed by IBM that combines many of the features of earlier languages, such as ALGOL, COBOL and FORTRAN. Many versions exist for microprocessors, such as PL/M, MPL, SM/PL, and  $PL\mu S.$
- PMOS P-channel metal-oxide semiconductor, a logic family that uses P-channel MOS field-effect transistors to provide high density and low speed.
- Pointer Register or memory location that contains an address rather than data.
- Polling Determining the state of peripherals or other devices by examining each one in succession.

Pop (or Pull) - Remove an operand from a stack.

- Port The basic addressable unit of the computer input/output section.
- Power-On Reset A circuit that automatically causes a RESET signal when the power is turned on, thus starting the system in a known state.
- Pre-empted Process A process which, because of scheduling policy, must relinquish the processor to another process.
- Printed Circuit Board (PC Board) A circuit board in which the connections are made by etching with a mask.
- Priority Interrupt System An interrupt system in which some interrupts have precedence over others--that is, will be serviced first or can interrupt the others' service routines.

Procedure-Oriented Language - See High-Level Language.

- Program A sequence of instructions properly ordered to perform a particular task.
- Program Counter A register that specifies the address of the next instruction to be fetched from program memory.
- Program Library A collection of debugged and documented programs.
- Programmable Interface An interface device that can have its active logic structure varied under program control.
- Programmable Logic Array (PLA) An array of logic elements that can be programmed to perform a specific logic function; like a ROM except that only certain addresses are decoded.
- Programmable Timer A device that can provide various timing modes and intervals under program control.
- Programmed Input/Output (I/O) Input/output performed under program control without using interrupts or direct memory access.
- PROM Programmable read-only memory, a memory that cannot be changed during normal operation but that can be programmed by the user under special conditions. The programming is generally not reversible.
- PROM Programmer A piece of equipment that is used to change the contents of a PROM.
- Prototyping System (or Development System) A hardware system used to breadboard a computer-based product. Contains the computer plus the software and hardware required for efficient development.
- Pseudo-Operation (or Pseudo-Instruction) An assembly language operation code that directs the assembler to perform some active but does not result in a machine language instruction.

Pull - See Pop.

Pullup Resistor - A resistor connected to the power supply that ensures that an otherwise open circuit will be at the voltage level of the power supply.

Pulse Generator - A device that produces a single pulse or a series of pulses of predetermined length in response to an input signal.

Push - Enter an operand into a stack.

Queue (or FIFO) - A set of registers or memory locations that are accessed in a first-in, first-out manner. That is, the first data entered into the queue will be the first data read.

- RAM Random-access (read/write) memory, a memory that can be both read and altered (written) in normal operation.
- Random Access All internal storage locations can be accessed in the same amount of time.
- Real Time In synchronization with the actual occurrence of events.
- Real Time System A real-time system is one which reacts to the environment by receiving data, processing it, and returning results sufficiently quickly to affect the reaction to the environment at that time.
- Real Time Clock A device that interrupts a CPU at regular time intervals.
- Recursive Subroutine A subroutine that calls itself as part of its execution.
- Re-entrant Subroutine A subroutine that can be executed correctly even while the same routine is being interrupted or otherwise held in abeyance.
- Refresh The process of restoring the contents of a dynamic memory before they are lost.
- Register A storage location used to hold bits or words inside the CPU.
- Register Direct Addressing An addressing method that is the same as direct addressing except that the address is in a register rather than in a memory location.
- Relative Addressing An addressing method in which the address specified in the instruction is the offset from a base address. The base address may be the contents of the program counter or a base register. Relative addressing allows programs to be easily relocated in memory.

and the second second

Relocatable - Can be placed in any part of memory without changes--that is, a program that can occupy any set of consecutive memory addresses.

Reset - A signal that starts a system in a known state.

- Resident Software Software that can run on the computer itself, unlike cross-assemblers or cross-compilers, which must run on another computer.
- Ripple Carry Forming the carry bit from a binary addition by using the carries from each bit position.
- ROM Read-only memory, a memory that contains a fixed pattern of data permanently defined as part of the manufacturing process.
- ROM Simulator A device that allows read/write memory to act like ROM during system development; the simulator usually has special display and debugging features.

Routine - A program or subprogram.

- RS Flip-Flop A flip-flop that can be placed in the 1 state by a signal on the SET input or in the 0 state by a signal on the RESET input.
- RS-232 A standard interface for the transmission of serial digital data.
- Schmitt Trigger A circuit used to produce a single, sharp transition (i.e., a pulse) from a slowly changing input.

Schottky TTL - A high-speed variant of standard TTL.

- Scratch-Pad Memory Memory locations or registers that are used to store temporary or intermediate results.
- Second Source A manufacturer who supplies a device or product originated by another manufacturer.
- Self-Assembler An assembler that runs on the computer for which it assembles programs.
- Self-Checking Number A number in which some of the digits serve to check for possible errors in the other digits and do not contain any additional information.
- Self-Test A procedure whereby a system checks the correctness of its own operation.

Separate I/O - Uses different lines for input and output.

·

Sequencer - A device that controls the ordering in time of the states of a system or the order in which instructions are executed.

Serial - One bit at a time.

Serial-Access - A storage device (such as a magnetic tape) from which data can only be reached or retrieved by passing through all intermediate locations between the desired one and the currently available one.

Set - Make state a logic one.

- Setup Time The time, prior to a clock transition, during which data must be stable for proper operation.
- Seven-Segment Code The code required to represent decimal digits or other characters on a seven-segment display.
- Seven-Segment Display A display made up of seven separately controlled elements that can represent decimal digits or other characters.
- Shift Register A clocked device that moves its contents one bit to the left or right during each clock cycle.
- Sign Bit The most significant bit of a register or memory location; a status bit that is one if the most significant bit of the result of the previous operation was one.
- Sign Extension The result of a right arithmetic shift that copies the sign bit into the succeeding less significant bits.
- Sign-Magnitude Number A number in which the most significant bit represents the sign or polarity and the remaining bits represent the magnitude.
- Signal Conditioning Making a signal compatible with the input requirements of a particular device through buffering, level translation, amplification, etc.
- Signature Analysis A method whereby faults can be found in bus-oriented digital systems by examining the time histories of signals at particular nodes.
- (Software) Simulator A computer program that follows the actions of a system in detail and that can be used for debugging or testing.
- Sink Current The ability of a device to accept current from external loads.

Small-Scale Integration (SSI) - An integrated circuit with a complexity of ten gates or less.

- Snapshot Record of the entire state of a system at a particular point in time.
- (Computer) Software Computer programs.
- Software Interrupt See Trap.
- SOS Silicon-on-sapphire, a faster MOS technology that uses an insulating sapphire substrate.
- Source Program Computer program written in an assembly or high-level language.
- Space The zero state on a serial data communications line.
- SPDT Switch Single-pole, double-throw switch with one common line and two output lines.
- SPST Switch Single-pole, single-throw switch with one common line and one output line.
- Stack A sequence of registers or memory locations that are used in a last-in, first-out manner--that is, the last data entered is the first to be removed and vice versa.
- Stack Addressing An addressing method whereby the data to be used is in a stack.
- Stack Frame A contiguous data area allocated for every activation of a routine; holds parameter values, local variables, temporary variables and return linkage information.
- Stack Pointer A register or memory location that is used to address a stack.
- Stand-Alone System A computer system that does not require a connection to another computer.
- Standard Teletypewriter A teletypewriter that operates asynchronously at a rate of ten characters per second.
- Standby (or Quiescent) Power The amount of power required to maintain the contents of a memory when it is not being accessed.
- Start Bit A one-bit signal that indicates the start of data transmission by an asynchronous device.
- State Counter A counter that contains the number of states that have occurred in the current operation.

Static Memory - A memory that does not change its contents without external causes, opposite of dynamic memory.

Status Bit - See Condition Code.

- Status Register (or Status Word) A register whose contents reflect the current status of the computer; may be the same as condition code register.
- Stop Bit A one-bit signal that indicates the end of data transmission by an asynchronous device.
- Strobe A one-bit signal that identifies or describes another set of signals and that can be used to clock or enable a register.
- Structured Programming A programming method whereby all programs consist of structures from a limited but complete set; each structure should have a single entry and a single exit.
- Subroutine A subprogram that can be reached from more than one place in a main program. The process of passing control from the main program to a subroutine is a "Subroutine Call" and the mechanism is a "Subroutine Linkage." The data and addresses that the main program makes available to the subroutine are "Parameters," and the process of making them available is called "Passing Parameters."

Subroutine Call - See Subroutine.

- Synchro-To-Digital Converter A device that converts an analog angle to a corresponding digital value.
- Synchronous Operation Operating according to an overall timing source, i.e., at regular intervals.
- Synchronization Making two signals operate according to the same clocking signal.
- Syntax The rules governing sentence or statement structure in a language.

Teleprinter - See Teletypewriter.

- Teletypewriter (TTY) A device containing a keyboard and a serial printer that is often used in communications and with computers.
- Terminal An input/output device at which data enters or leaves a computer system.

- Time-Shared Bus A bus that is used for different purposes at different times.
- Top-Down Design A design method whereby the overall structure is designed first and parts of the structure are subsequently defined in greater detail.
- Trap An instruction that forces a program to jump to a specific address, often used to produce breakpoints or to indicate hardware or software errors.
- Tri-State (or Three-State) Logic outputs with three possible states--high, low, and an inactive (high-impedance or open-circuit) state that can be combined with other similar outputs in a busing structure.
- Tri-State Enable (or Select) An input that, if not active, forces the outputs of a tri-state device into the inactive or open-circuit state.
- TTL (Transistor-Transistor Logic) The most widely used bipolar technology for digital integrated circuits. Popular variants include high-speed Schottky TTL and low-power Schottky (or LS) TTL.
- TTL-Compatible Uses voltage levels that are within the range of TTL devices and can be used with TTL devices without level shifting, although buffering may be necessary.
- 2-Key Rollover (2KRO) Resolving two (but not more) simultaneous key closures into two consecutive output codes.
- Two's Complement A binary number that, when added to the original number in a binary adder, produces a zero result. The two's complement is the one's complement plus one.
- Two's Complement Overflow A situation in which a signed arithmetic operation produces a result that cannot be represented correctly--that is, the magnitude overflows into the sign bit.
- Unbundling Pricing certain types of software and services separately from the hardware.
- ULA Uncommitted Logic Array.
- Universal Asynchronous Receiver/Transmitter (UART) An LSI device that acts as an interface between systems that handle data in parallel and devices that handle data in asynchronous serial form.

- Universal Synchronous Receiver/Transmitter (USRT) An LSI device that acts as an interface between systems that handle data in parallel and devices that handle in synchronous serial form.
- Urgency The degree to which a process requires attention; determined by the processes' priority.
- Utility Program A program that provides basic functions, such as loading and saving programs, initiating program execution, observing and changing the contents of memory locations, or setting breakpoints and tracing.

UVPROM (or UVROM) - See EPROM.

- Vectored Interrupt An interrupt that provides the CPU with an identification code that the CPU can use to transfer control to the corresponding service routine.
- VHSIC Very-High-Speed-Integrated Circuits, project of DOD to develop VLSI signal processors.
- Very-Large-Scale Integration (VLSI) An integrated circuit (similar to LSI) with complexity equivalent to over 100,000 gate RAM chips by 1980 and million-gate chips by the year 2000.
- Volatile Memory A memory that loses its contents when power is removed.
- Wired-OR Connecting outputs together without gates to form a busing structure; requires special outputs of which only one is active at a time.
- Word The basic grouping of bits that the computer can manipulate in a single cycle.
- Word Length The number of bits in the computer word, usually the length of the computer's data bus and data and instruction registers.

Working Register - See General-Purpose Register.

Zero Bit - A status bit that is one if the last operation produced a zero result.

## LIST OF REFERENCES

- Bhattacharyya, J.C., "Criteria for Selection of Microprocessor for Military Applications," <u>IETE</u>, V. 24, No. 3/4 March/April 1978.
- 2. Leventhal, L.A., <u>Introduction to Microprocessors:</u> <u>Software, Hardware Programming</u>. Prentice-Hall, 1978.
- 3. Burr, W.E. and Coleman, A.H. (eds.), "Final Report of the CFA Selection Committee," <u>ECOM</u>, September 1977.
- 4. Eckhouse, R.H., Jr. and Morris, L.R., <u>Minicomputer</u> <u>Systems, Organization, Programming and Applications</u> (PDP-11). Prentice-Hall, 1979.
- 5. Korn, G.A., <u>Microprocessors and Small Digital Computer</u> Systems for Engineers and Scientists. McGraw-Hill, 1977.
- Gale, E.G. and Gremillion, L.L., "Price/Performance Patterns of U.S. Computer Systems," <u>ACM</u>, V. 22, No. 22, April 1979.
- 7. Madnick, E.S. and Donovan, J.J., <u>Operating Systems</u>. McGraw-Hill, 1974.
- Schell, R.R., "Computer Security: The Achilles' Heel of the Electronic Air Force?" <u>Air University Review</u>, January-February 1979.
- 9. <u>Electronic Design</u>, "Bubbles--More Capacity, Components," Vol. 27, No. 24, 22 November 1979.
- 10. Prokop, J., <u>Computers in the Navy</u>, Naval Institute Press, 1976.
- 11. Burr, E.W. and Smith, R.W., "Comparing Architectures," Datamation, February 1977.
- 12. Johannsen, C., "Choosing Software," <u>Computerworld</u>, 28 January and 4 February 1980.
- 13. Bernhard, R., "Computers: Emphasis on Software," IEEE Spectrum, January 1980.
- Snigier, P., "Magnetic Bubble Memories," <u>Digital Design</u>, December 1979.

- 15. Computerworld, "Multitasking Business Micro Handles 48 Users," January 28, 1980, p. 67.
- 16. Reghizzi, C.S., Corti, P., and Darpa, A., "A Survey of Microprocessor Languages," Computer, January 1980.
- 17. Glass, L.R., "Ada Programming Language Gets Fine-Tuning," Computerworld, November 19, 1979, p. 21.
- 18. Hurwitz, J. and Klunucan, P., "Mini-Micro World," <u>Mini-Micro Systems</u>, December 1979, p. 35.
- 19. Bursky, D., "Microprocessor Data Manual," <u>Electronic</u> Design 24, 22 November 1979.
- 20. Cushman, H.R., "µP/µC Chip Directory," EDN, 20 October 1979.
- 21. Newman, M.W. and Sproull, F.R., <u>Principles of Inter-</u> active Computer Graphics, McGraw Hill, 1979.
- 22. Orr, J., "Interactive Computer Graphics Systems," <u>Mini-Micro Systems</u>, December 1979.
- 23. Weitzman, C., <u>Distributed Micro/Minicomputer Systems</u>, Prentice-Hall, 1980.
- 24. <u>Computerworld</u>, "Japanese Ready Semi Industry for VLSI," 24 March 1980, p. 59.
- 25. Christiansen, D., "Technology '80," <u>IEEE Spectrum</u>, January 1980, p. 30.
- 26. Nyman, H.T. and Hazan, P., "Dedicated Microcomputers for Defense and Industry," Computer, February 1979, pp. 85-88.
- 27. Auerbach Publishers, Inc., "Checklist for Evaluating Minicomputer Vendor Policies," (2-04-11), 1977.
- 28. Lucas, C.H., Jr., "Performance Evaluation and Monitoring," <u>Computer Surveys</u>, Vol. 3, No. 3, September 1971.
- 29. Summey, W.L., "VLSI with a Vengeance," IEEE Spectrum, April 1980, p. 24.
- 30. Kodres, U.R., Hamming, W.R., Buttinger, D.J. Jones, R.C., <u>A Study of Alternatives for VSTOL Computer Systems</u>, Naval Postgraduate School, Monterey, CA, NPS52-78-001, April 1978.

- 31. NAVAIR 01-85 ADF-2-10.1.1 Technical Manual, <u>Navy Model</u> <u>A-6E Aircraft. Integrated Weapons System Theory</u>, NAVAIR, 1 September 1971.
- 32. Hirsh, P., "Development of Ada Gets Mixed Reviews," <u>Computerworld</u>, 26 May 1980, p. 20.
- 33. Palmer, J., Nave, R., Wymore, C., Koehler, R., and McMinn, C., "Making Mainframe Mathematics as Accessible to Microcomputers," <u>Electronics</u>, 8 May 1980, pp. 114-121.

the states of

## INITIAL DISTRIBUTION LIST

		No.	Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314		2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940		2
3.	Department Chairman, Code 52 Department of Computer Science Naval Postgraduate School Monterey, California 93940		1
4.	Professor Uno Kodres, Code 52 Ko Department of Computer Science Naval Postgraduate School Monterey, California 93940		1
5.	Professor B. J. Carey, Code 52 Ca Department of Computer Science Naval Postgraduate School Monterey, California 93940		2
6.	Hellenic Navy General Staff Education Department/3 Stratopedon Papagou Holargos, Athens, GREECE		3
7.	LCDR M. Mastrakas HN Asty Egyptioton 4A Kifisia, Athens, GREECE		5
8.	LCDR A. Koutsotolis HN SMC 2738 Naval Postgraduate School Monterey, California 93940		3
9.	Captain S.Tsavdaris HA SMC 2819 Naval Postgraduate School Monterey, California 93940		2

