INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION

INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
34100 TRIESTE (ITALY) - P.O.B. 586 - MIRAMARE - STRADA COSTIERA 11 - TELEPHONES: 224281/2/3/4/5/6
CABLE: CENTRATOM - TELEX 460392-1

SMR/90 - 31

COLLEGE ON MICROPROCESSORS:

TECHNOLOGY AND APPLICATIONS IN PHYSICS

7 September - 2 October 1981

DESK-TOP COMPUTERS AND FLOATING-POINT PROCESSORS

C. VERKERK

DD Division
CERN
1211 Geneva 23
Switzerland

Desk-Top Computers and Floating-Point Processors.

Desk-Top Computers?

Equivalent to: Hobby, Home, Personal } Computers

Floating-Point Processors?

Equivalent to: Scientific Calculations "Number Crunchers

It is useful to investigate these topics, find out if these devices may help to solve your problems in Physics:

- Data acquisition and recording
- Monitoring
- Control
- Analysis of experimental data.

Warnings: These lectures do not Provide:

- General solutions to problems
- Recipes
- Catalogue of Products

Disjoint set of topics:

- Floating-Point Calculations
- Arithmetic Chips
- Examples
- Fast multiplier chips
- Home Computers:
  - Characteristics
  - Languages
  - Operating Systems
- Extensions to Home Computers
  - Hardware
  - Software
- Personal Computers for Professionals.
  - Principle
  - Examples: Alto, Apollo, Lilith
  - Local Area Network

BUY OR BUILD?

ADVANTAGES OF BUILDING:

- CHEAP (IN COMPONENTS)
- SYSTEM CAN BE OPTIMIZED
- EDUCATIONAL
- GREAT FUN

DISADVANTAGES:

- SOFTWARE EFFORT WILL BE LARGE
- SAME FOR DOCUMENTATION
- RISK OF A ONE-OFF PRODUCT (VERY SPECIAL PURPOSE, NOT ADAPTABLE)
- TOOLS MAY BE EXPENSIVE
- IT WILL TAKE MUCH TIME, SO TOTAL COST MAY BE HIGH
- MAY BE FRUSTRATING IN THE END

THE ANSWER TO THE PROBLEM WILL DEPEND ON LOCAL CIRCUMSTANCES:

- RELATIVE WEIGHT OF THE ARGUMENTS
- AVAILABILITY OF EQUIPMENT
- AVAILABILITY OF EXPERTISE
- TEACHING REQUIREMENTS

REQUIREMENT SPECIFICATION IS IMPORTANT

CAN WE TURN A MICROPROCESSOR INTO A NUMBER CRUNCHER?

DIFFICULT:

- LIMITED PRECISION

  8-BIT: 0 TO 255, OR -128 TO 127

  16-BIT: 0 TO 65536, OR -32768 TO 32767

  MULTIPLE PRECISION NECESSARY

- MICROPROCESSORS CANNOT MULTIPLY

  NEED FOR EITHER A HARDWARE EXTENSION OR SOFTWARE ROUTINES

- MICROPROCESSORS ARE TOO SLOW

  INHERENT SPEED IS FINE, BUT THE FIRST TWO LIMITATIONS MAKE THEM SLOW FOR NUMERICAL CALCULATIONS

MULTIPLICATION.

DECIMAL:

```
  325
  647   *
 2275
1300.
1950..
------  +
210275
```

NOTE:
- TABLES OF MULTIPLICATION
- SHIFT AND ADD
- RESULT HAS MANY DIGITS

BINARY:

1) UNSIGNED

- TABLE OF MULTIPLICATION:

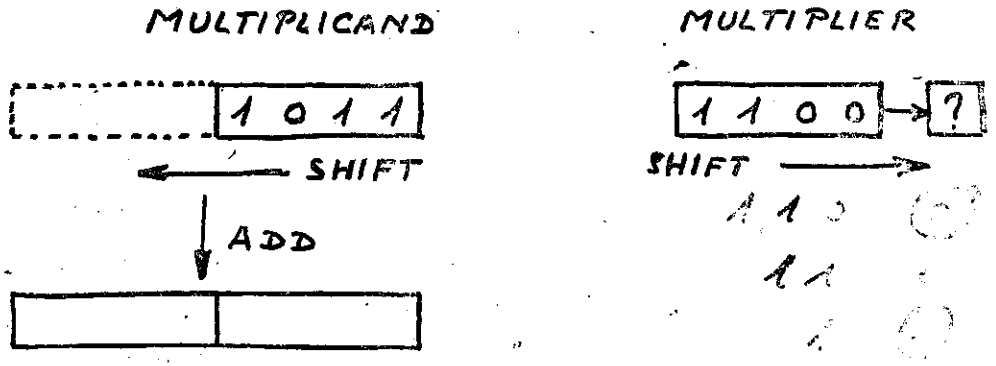
```
0 * 0 = 0
0 * 1 = 0
1 * 0 = 0
1 * 1 = 1
```

THE SAME AS THE BOOLEAN AND FUNCTION

```
 1011          11
 1100    *     12    *
------         --
 0000          22
 0000.        11.
 1011..       ---
 1011...      132
--------  +
10000100  = 8*16 + 4 = 128 + 4 = 132
```


- SHIFT AND ADD:
  - RESET RESULT TO ZERO
  - SHIFT MULTIPLIER RIGHT (1 BIT)
  - INSPECT THE MULTIPLIER BIT JUST SHIFTED OUT.

    IF IT IS A 1
    THEN ADD THE (SHIFTED) MULTIPLICAND TO THE RESULT
    ELSE ADD ZERO TO RESULT (OR DO NOTHING)
  - SHIFT THE MULTIPLICAND LEFT (1 BIT)
  - REPEAT AS MANY TIMES AS THERE ARE BITS IN THE MULTIPLIER



- DOUBLE PRECISION:

			cycles
ADD :	ADD B	$MCAND_{LOW}$	4
	ADC A	$MCAND_{HIGH}$	4
LEFT SHIFT :	ROL	$MCAND_{LOW}$	6
	ROL	$MCAND_{HIGH}$	6

CAREFUL WITH THE CARRY!

BODY OF THE LOOP TAKES ≈ 30 CYCLES.
THUS 8×8 BIT MULTIPLICATION WILL TAKE
≈ 8 × 30 × 1 μs = 240 μs

16×16 BIT IS MUCH WORSE, BECAUSE INTERMEDIATE RESULTS CANNOT BE KEPT IN THE ACCUMULATORS, BUT MUST BE STORED AND PICKED UP THE NEXT TIME ROUND.
THIS ADDS AT LEAST 2 × 4 × 3 = 24 CYCLES TO THE BODY OF THE LOOP, FOR THE ADDITIONS.
FOR THE SHIFTS THE SAME!
≈ 16 × 100 × 1 μs = 1600 μs FOR 16×16 BIT!

2) SIGNED MULTIPLICATION.

— 2's COMPLEMENT REPRESENTATION AVOIDED THE NEED FOR A SUBTRACTOR, DISTINCT FROM THE ADDER.

MULTIPLICATION CAN BE DONE IN 2's-COMPLEMENT, BUT IS MESSY.

$$\text{VALUE} = -B_{n-1}.2^{n-1} + \sum_{k=0}^{n-2} B_k.2^k = a + b$$

$$(a+b)(c+d) = ac + ad + bc + bd$$

ac: SIGN; ad + bc: THESE TERMS MAY HAVE TO BE SUBTRACTED; bd: UNSIGNED

— MULTIPLICATION OF SIGNED INTEGERS IS STRAIGHTFORWARD IF SIGN-MAGNITUDE REPRESENTATION IS USED

DIVISION.

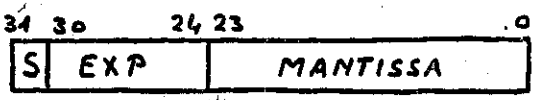
· IN PRINCIPLE BASED ON SHIFT AND SUBTRACT ALGORITHMS. DETAILS ARE COMPLICATED AND MESSY.

FLOATING-POINT REPRESENTATION.

EXTENDS THE RANGE OF NUMBERS THAT CAN BE REPRESENTED, NOT THEIR PRECISION.

EQUIVALENT TO SCIENTIFIC NOTATION:
1.38×10^{6} ; -0.7391×10^{-13}

IN BINARY:
$(-1)^{S} \times \text{MANTISSA} \times 2^{\text{EXP}}$



THIS IS THE GENERAL FORMAT.
THERE ARE AT LEAST AS MANY DETAILED FORMATS AS THERE ARE TYPES OF COMPUTERS (SINGLE AND DOUBLE PRECISION!).

IN GENERAL: $-64 \longrightarrow +63$ (0 – 127)
EXP = TRUE EXPONENT + BIAS

— THIS MAKES IT POSSIBLE TO HAVE NEGATIVE EXPONENT AND TO REPRESENT NUMBERS < 1

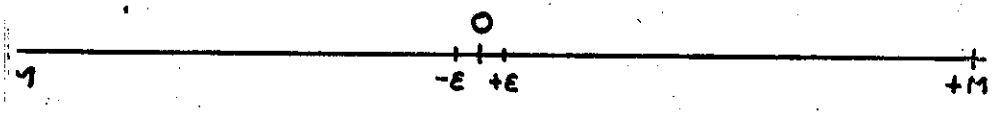
Some Idiosynchrasies:

- S together with mantissa may be:
  - Sign and magnitude
  - 2's complement
  - 1's complement
- Mantissa may be
  - Integer (binary point at the right of the word)
  - Fraction (binary point at the left of the mantissa)

  (A fraction $= B_{23}\cdot 2^{-1} + B_{22}\cdot 2^{-2} + \ldots + B_0\cdot 2^{-24}$)
- Mantissa may be
  - Normalized (MSB = most significant bit = 1)
  - Unnormalized
- Exponent may be
  - Biased
  - 2's complement number
  - 1's complement number
- Exponent may be the power to which you should raise
  - 2
  - 16 (IBM 370 series!)
- Number of bits in the fields differs from machine to machine

What a mess!

Range and Precision



Machine	Range ε	Range M	Relative Precision
DPP11, VAX single prec.	0.29×10^{-38}	1.7×10^{38}	2^{-24}
DPP11, VAX double prec.	0.29×10^{-38}	1.7×10^{38}	2^{-56}
IBM 370 short	5.4×10^{-79}	7.2×10^{75}	2^{-24}
IBM 370 long	5.4×10^{-79}	7.2×10^{75}	2^{-56}
INTEL 8087 short	1.2×10^{-38}	3.4×10^{38}	2^{-24}
INTEL 8087 long	2.2×10^{-308}	1.8×10^{308}	2^{-53}

The advantage of floating-point calculations is that you do not have to worry anymore about the size of the numbers and the conceptual position of the binary point.

Floating-Point Operations

$$V_1 = (-1)^{S_1} \cdot M_1 \cdot 2^{E_1}$$

$$V_2 = (-1)^{S_2} \cdot M_2 \cdot 2^{E_2}$$

Then **multiplication**:

$$V_1 \times V_2 = (-1)^{S_1 + S_2} \cdot (M_1 \times M_2) \cdot 2^{(E_1 + E_2)}$$

Thus: Result has

$S_R = (S_1 + S_2)$ modulo 2 $\quad (1+1 \rightarrow 0)$

$M_R = M_1 \times M_2$

$E_R = E_1 + E_2$

(if E = true exponent + bias, then $E_R = E_1 + E_2 - \text{bias}$)

Division is similar.

Addition cannot be done without a preliminary renormalization which makes the two exponents equal.

Suppose $E_1 = E_2 + K > E_2$

Then:

$$V_2 = (-1)^{S_2} \cdot (M_2 \times 2^{-K}) \cdot 2^{E_2 + K}$$

Before performing the addition, you must for the number with the smaller exponent: i) Shift Mantissa Right

ii) Add one to Exponent

iii) Repeat until exponents equal

Floating-Point Attachments

A floating-point unit is complicated.

On the market: Am 9511A

Am 9512

Intel 8087

Am 9511A is called an Arithmetic Processing Unit (APU).

It does several things:

- Fixed-Point Arithmetic on 16 or 32 bit (Add, Subtract, Multiply, Divide).
- Floating-Point Arithmetic on 32-bit
- Trigonometric Functions: SIN, COS, TAN, ARCSIN, ARCCOS, ARCTAN.
- Exponentials and Logarithms: LN, e^x, $\text{Log}_{10} X$, X^Y.
- Conversion Fl-Pt $\rightleftharpoons$ Fix. Pt

- It is an internally microprogrammed stack computer.
- Data is exchanged one byte at a time, and put on, or taken from top of stack.
- Commands are 1 byte.

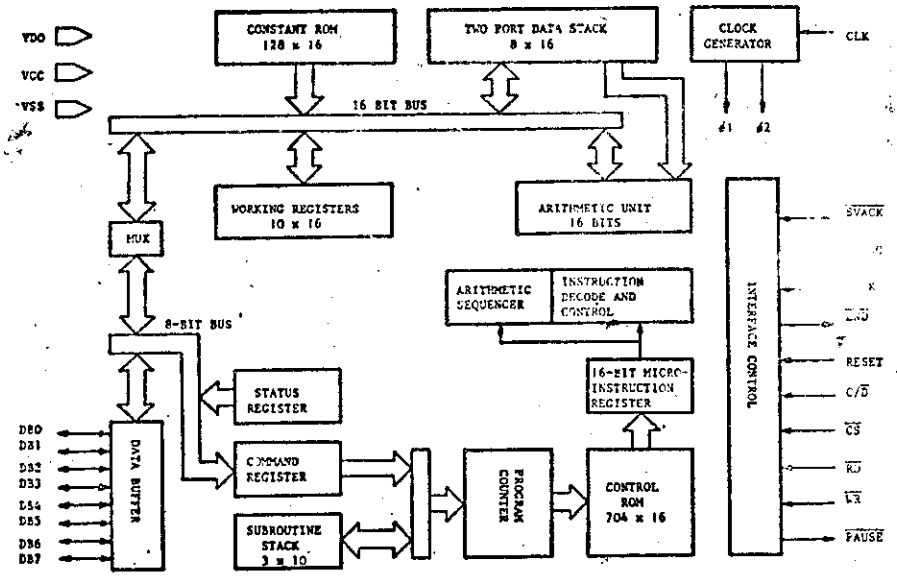
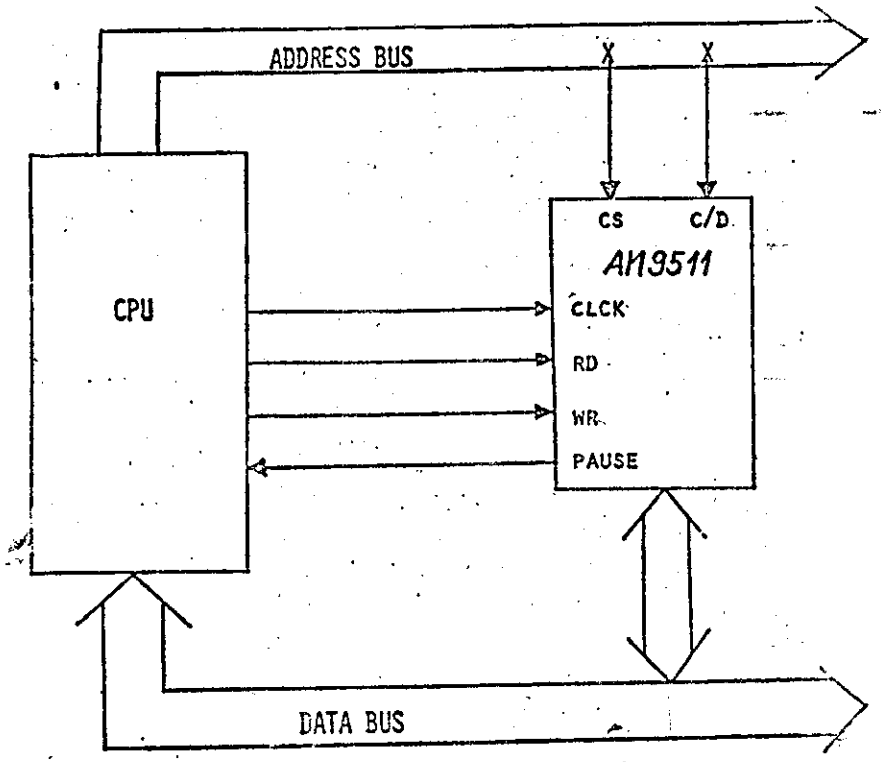
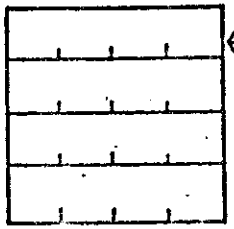


Figure 1. Arithmetic Processing Unit Block Diagram.

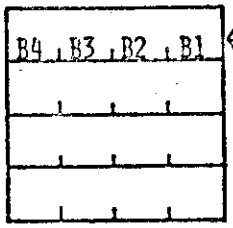


STACK

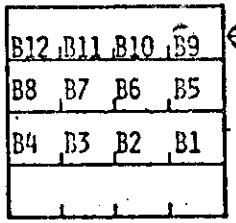
INITIAL STATE



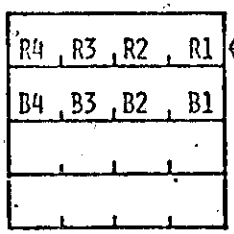
AFTER INPUT OF 1.ST 4 BYTES. B1 IS LSB AND ENTERED FIRST



AFTER INPUT OF TWO MORE 32-BIT OPERANDS



AFTER COMPLETION OF FIRST OPERATION R4 IS LSB.

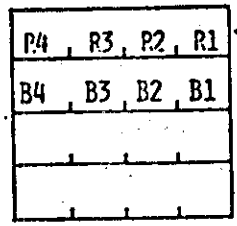


$P = b * c$

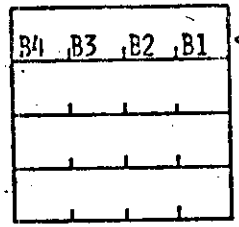
$S = a + d$

ung

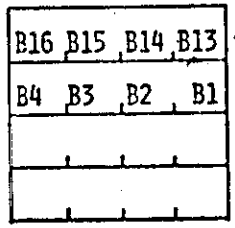
AS BEFORE, READY FOR READING. R4 WILL BE READ FIRST



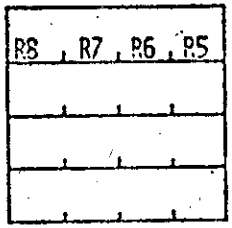
AFTER READING THE RESULT



AFTER INPUT OF A NEW OPERAND



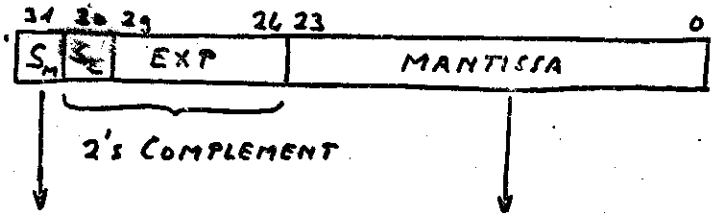
A NEW RESULT HAS BEEN OBTAINED



$P = b * c$

$S = a + d$

DATA FORMAT:



SIGN $\quad 0.5 \leq$ MAGNITUDE $< \approx 0.999\,9999$

RANGE: 2.71×10^{-20} TO 9.22×10^{18}

STATUS WORD:

MSB 7	BUSY	
6	TOS < ϕ	
5	TOS = 0	
4	ERROR	01 NEGATIVE ARGUMENT
3		10 ZERO DIVISOR
		11 ARGUMENT TOO LARGE
2	UNDERFLOW	
1	OVERFLOW	
LSB 0	CARRY OR BORROW	

COMMAND WORD:

MSB 7	SERVICE REQUEST:	ϕ SVREQ WILL REMAIN LOW
		1 SVREQ HIGH AT END OF OP.
6	DATA FORMAT:	ϕϕ : 32-BIT FL. PT.
5		ϕ1 : 32-BIT FIXED PT.
		1ϕ : UNDEFINED
4	OPERATION CODE	11 : 16-BIT FIXED PT.
3		
2		
1		

OPERATION:

- LOAD DATA, STARTING WITH LEAST SIGNIFICANT BYTE OF FIRST OPERAND
- LOAD COMMAND

  9511 EXECUTES REQUESTED OPERATION
- READ RESULT

THIS IS BEST DONE BY A SUBROUTINE.
PARAMETERS TO BE PASSED:

- ADDRESSES (OR VALUES) OF OPERANDS
- AN INTEGER REPRESENTING THE OPERATION CODE. THERE ARE TWO WAYS OF DOING THIS:
  - THE INTEGER ≡ OPERATION CODE
  - INTEGER = PARAMETER FOR A CASE STATEMENT IN THE SUBROUTINE.
- POSSIBLY A PARAMETER SPECIFYING DATA FORM

WHAT IS THE SUBROUTINE GOING TO DO WHEN 9511 IS BUSY?

POSSIBILITIES:

- READ STATUS AND CHECK BUSY BIT. SLOWS DOWN THE APU.
- HAVE THE MICROPROCESSOR SUSPEND EXECUTION (USE $\overline{\text{PAUSE}}$ FOR THIS)
- GO AND DO SOMETHING ELSE, UNTIL INTERRUPT. (USE END SIGNAL TO GENERATE IT)
- EXECUTE WAI INSTRUCTION (6800)

PERFORMANCE:

OPERATION	TIME (µs)	OPERATION	TIME (µs)
SADD	8		
SSUB	13	SIN	1882
SMUL	41	COS	1856
SDIV	43	TAN	2408
DADD	10	ARC SIN	3038
DSUB	19	ARC COS	3151
DMUL	69	ARC TAN	2492
DDIV	69	LN	1908
FADD	28-168	e^x	1955
FSUB	35-169	LOG_{10}	1968
FMUL	57	x^y	3863
FDIV	57		

Am 9512:

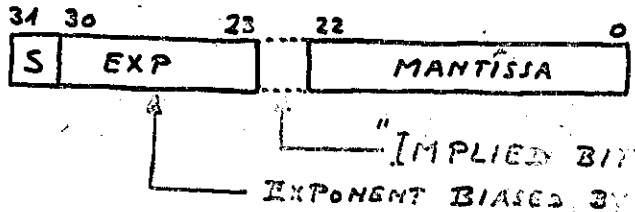
+ FLOATING-POINT ARITHMETIC ONLY
- COMPATIBLE WITH PROPOSED STANDARD
- SINGLE (32-BIT) AND DOUBLE (64-BIT) PRECISION

PERFORMANCE OF 9512:

- FOR 32-BIT ≈ 9511
- FOR 64-BIT:

		COMPARED WITH 8087
ADD	276 - 1235 µs	< 1
SUB	253 - 1478 µs	< 1
MUL	768 - 863 µs	< 1
DIV	2043 - 2319 µs	< 1

PROPOSED STANDARD:



$$\text{VALUE} = (-1)^S \times 2^{\{E-(2^7-1)\}} \times (1 \cdot M)$$

INTEL 8087

- COPROCESSOR FOR THE 8086
- STACK COMPUTER
- CAN ALSO ACCESS MEMORY
- CAN ACCESS OTHER STACK ELEMENTS
- INTERNAL 80-BIT FORMAT (TEMPORARY REAL)
- INTEGERS: 16, 32 AND 64-BIT
- FLOATING POINT: 32, 64 AND 80-BIT
- PACKED DECIMAL (80 BITS, 18 DIGITS AND SIGN)
- SUBTRACT AND SUBTRACT REVERSE
- SAME FOR DIVIDE.
- COMPARE INSTRUCTIONS
- SCALE, DECOMPOSE, NEGATE, ABSOLUTE VALUE
- SQRT, TAN, ARCTAN, EXPONENTIAL, LOGARITHM.
- ON-CHIP EXCEPTION HANDLING

PERFORMANCE:

COMPARE	6	μs	
ADD (MAGNITUDE)	10	μs	
SUBTRACT (MAGNITUDE)	16	μs	
MULTIPLY	MAX. 24	μs	(16 FOR SHORT REAL)
DIVIDE	38	μs	
SQUARE ROOT	38	μs	

A VERY HIGH PERFORMANCE FLOATING-POINT PROCESSOR!

AN EXAMPLE FROM HIGH-ENERGY PHYSICS.

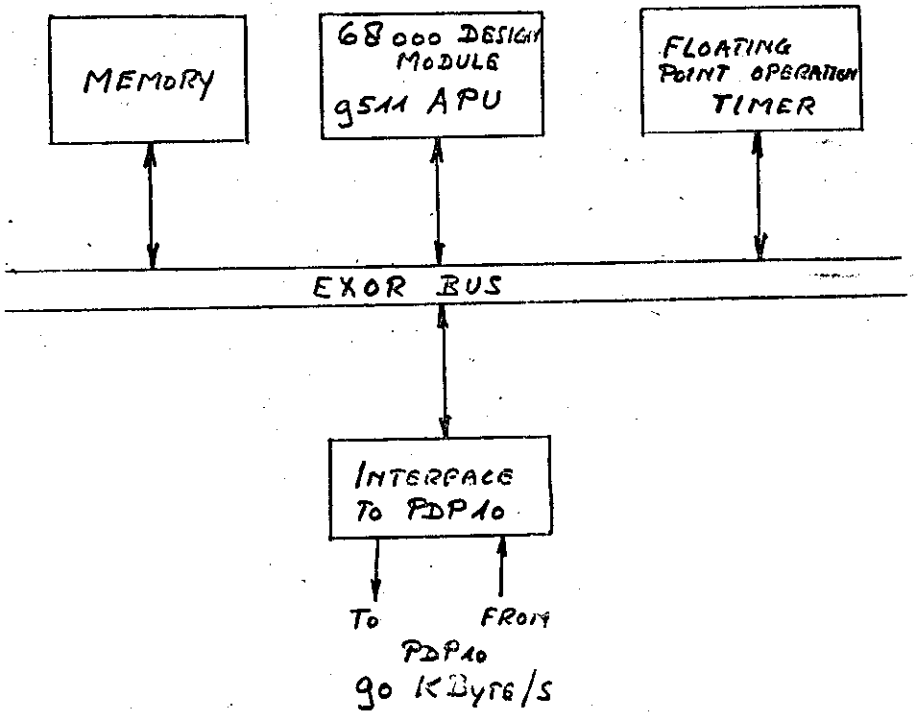
AT BROOKHAVEN NATIONAL LABORATORY AN EXPERIMENTAL PROCESSOR WAS BUILT TO RUN PATTERN RECOGNITION CODE FOR EVENTS IN THE MULTIPARTICLE SPECTROMETER.

IT CONSISTED OF MOTOROLA 68000 AND Am 9511

A FORTRAN COMPILER WAS DEVELOPED FOR THE 68000. AS IT WAS EXPECTED THAT HUNDREDS OF KILOBYTES OF EXISTING FORTRAN PROGRAMS WOULD RUN ON THE PROCESSOR.

PERFORMANCE MEASUREMENTS WERE MADE USING A PARTICULAR ROUTINE.
THE RESULTS WERE EXTRAPOLATED TO
A 8 MHz 68000 (INSTEAD OF 4 MHz)
A NC16081 (INSTEAD OF 9511)

THE END RESULT:
SUCH A COMBINATION OF A POWERFUL 16-BIT MICROPROCESSOR AND A FLOATING-POINT ATTACHMENT HAS THE POWER OF
≈ 1/30 OF A CDC 7600.
≈ 1/4 OF A PDP10 (KA10)



BERNSTEIN ET. AL., A MICROPROCESSOR-BASED SINGLE BOARD COMPUTER FOR HIGH ENERGY PHYSICS EVENT PATTERN RECOGNITION, IN PROC. TOPICAL CONF. APPLICATION MICROPROCESSOR TO HEP EXPERIMENTS, MAY 1981, GENEVA; CERN 81-07. pp 479-487

THESE FLOATING-POINT ATTACHMENTS SHOULD NOT BE CONFOUNDED WITH

· FAST MULTIPLIER CHIPS.

THERE ARE SEVERAL ON THE MARKET:

- Am 25S05 (4x2, 76 ns) } TO BE PUT IN ARRAYS
- MC 10183 (4x2, 50 ns) }

- TRW 8 HJ	(8x8, 65 ns)	2's COMPL.
- TRW 8 HUJ	(8x8, 45 ns)	UNSIGNED MAGNITUDE
- TRW 12 HJ	(12x12, 80 ns)	2's COMPL + UNS. MAGN.
- TRW 16 HJ	(16x16, 100 ns)	2's COMPL + UNS. MAGN.
- Am 25S557	(8x8, 45 ns)	" "
- MMI 67558	(8x8, 80-100 ns)	" "
- MC 10901	(8x8, 19 ns)	" "

- MMI 67516	(16x16, 800 ns)	2's COMPLEMENT BUS ORIENTED
- Am 25LS2516	(8x8, 400 ns)	
- 74F559	(8x8, 320 ns)	
- SBP 9708	(8x8, 275 ns)	

THESE CHIPS ARE INTENDED FOR CONSTRUCTION OF FAST ARITHMETIC UNITS, AS NEEDED IN SIGNAL PROCESSING (FFT, CORRELATORS, FILTERS)

THE LAST GROUP CAN BE ATTACHED TO MICROPROCESSORS, BUT WITH CONSIDERABLE HARDWARE EFFORT.

THEY DO FIXED-POINT ONLY.

A FEW REFERENCES:

- Am. 9511 APU DATA SHEET, ADVANCED MICRO DEVICES INC, MARCH 1978
- R.O. PARKER, J.H. KROEGER, ALGORITHM DETAILS FOR THE Am 9511, PUBLICATION AM-PUB072, AMD Inc. 1978
- SEVERAL ARTICLES IN:
  - COMPUTER DESIGN:
    - APRIL 1981 (CROSON ET. AL.)
    - JUNE 1979 (GERBERICH ET. AL.)
    - JULY 1980 (B.K. GUPTA)
  - BYTE: 1(13), 1976, pp. 26-33 (G.R. SCOTT)
  - ELECTRONICS: 48, 1977, p. 113 (D.J. GUST)

ON THE INTEL 8087:

- J.F. PALMER, THE INTEL 8087 NUMERIC DATA PROCESSOR, IN PROC. NAT. COMP. CONF, 1980, p. 887
- ?, ? IN PROC. 7TH INTERNAT. SYMPOSIUM ON COMPUTER ARCHITECTURE, LA BAULE, FRANCE, MAY 1980. PUBLISHED BY ACM AND IEEE

- ON THE TRIAL 68000 + 9511 THERE IS AN ARTICLE IN PROC. TOPICAL CONF. ON APPLICATIONS OF MICROPROCESSORS TO HIGH-ENERGY PHYSICS EXPERIMENTS, GENEVA, MAY 1981. CERN 81-07

- FAST MULTIPLIERS: ELECTRONIC DESIGN NEWS APRIL 1981

PERSONAL COMPUTERS.

HARD TO DEFINE PRECISELY, BUT THEY ARE

- CHEAP (BASE PRICE < $2500)
- PROVIDED WITH SUFFICIENT
  - SOFTWARE
  - DOCUMENTATION

  SO THAT A LAYMAN MAY RAPIDLY PROFIT FROM ITS USE.
- GENERALLY SOLD IN "COMPUTER STORES" OR EVEN IN DEPARTMENT STORES.

TYPICALLY:

- GENERAL-PURPOSE
- STAND-ALONE
- MICROPROCESSOR-BASED
- CONVERSATIONAL INTERACTION WITH USER.

THE MICROPROCESSOR IS IN MOST CASES A ZILOG Z80, MOTOROLA 6800 OR A MOS-TECHNOLOGY 6502. ALL 8-BIT

MEMORY SIZE (RAM) RANGES FROM 4 KBYTES TO 64 KBYTES.

SYSTEM IS USUALLY SUPPLIED WITH A KEYBOARD.

PERIPHERALS THAT CAN BE EASILY ATTACHED:

- B/W TV MONITOR (SUPPLIED OR YOUR OWN)
- CASSETTE TAPE DRIVE
- 5¼ OR 8-INCH DISKETTE (FLOPPY DISK)

THERE IS NO MONOPOLY FOR HOME COMPUTERS, WHILE THE MARKET IS ENORMOUS. HUNDREDS (THOUSANDS?) OF SMALL FIRMS ARE ACTIVE IN THE FIELD.

ADVANTAGES FOR THE USER:

- HE CAN BUY CHEAPLY A WIDE RANGE OF COMPONENTS AND HARDWARE EXTENSIONS.
- HE CAN OBTAIN CHEAPLY MANY SOFTWARE PACKAGES.

FIRST HOBBY COMPUTER WAS ALTAIR 8800 (1975, MITS Inc., $ 400 IN KIT FORM)

THE ONES SELLING BEST:

		AT END 1979
RADIO SHACK's	TRS-80	150 000
COMMODORE	PET AND CBM	105 000
APPLE	APPLE II AND II PLUS	65 000
≈ 15 OTHERS		80 000
		400 000

LA REPUBBLICA 18 SEPTEMBER 1981:

TRS-80	31 %
COMMODORE	20.8 %
HEWLETT-PACKARD	20 %
IBM	5.3 %
APPLE	2.0 %

APPLICATIONS

- HOME / HOBBY COMPUTING
  - GAMES
  - MUSIC
  - TELEPHONE ANSWERING, DIALING
  - ENVIRONMENT / HOME APPLIANCE CONTROL
  - BUDGETING, INVESTMENT ANALYSIS
  - PERSONAL RECORDS, DIARIES
  - RECIPES
  - WORD PROCESSING
- EDUCATION
  - COMPUTER-ASSISTED INSTRUCTION (CAI)
- PROFESSIONAL AND BUSINESS
  - BUDGETING
  - GENERAL LEDGER
  - TAX / PAYROLL ACCOUNTING
  - PERSONAL RECORDS
  - FILING
  - CALENDARS
  - REPORT PREPARATION
  - WORD PROCESSING
  - ELECTRONIC MAIL

CLEAR TREND TO MOVE FROM FRIVOLOUS TO MORE SERIOUS APPLICATIONS. THIS IS SHOWN BY THE SYSTEMS CONFIGURATIONS AND SOFTWARE PACKAGES OFFERED.

INNOVATIONS

- PRICE/PERFORMANCE
- MARKETING, DISTRIBUTION AND MAINTENANCE
- DOCUMENTATION
- PERIPHERALS (VDU, FLOPPIES, PRINTERS)
- LANGUAGES
- OPERATING SYSTEMS FOR SMALL CONFIGURATIONS
- USE OF GRAPHICS
- EDITORS
- DATA COMMUNICATIONS

PROBLEMS MANUFACTURERS ARE FACING:

- INFORMATION FOR THE POTENTIAL CUSTOMER
- TRANSPORTABILITY OF SOFTWARE
- SOFTWARE PIRACY

TRANSPORTABILITY IS MADE DIFFICULT BY HARDWARE DESIGN CHOICES:
- MEMORY MAP (RAM/ROM)
- ADDRESSES AND CONVENTIONS FOR I/O

TO PREVENT SOFTWARE PIRACY PROTECTION MECHANISMS ARE USED:

- HARDWARE: KEYS WIRED IN THE SYSTEM
  PLAs FOR ADDRESS TRANSLATION
- SOFTWARE: SCRAMBLED PROGRAMS

LANGUAGES

BASIC IS MOST WIDELY USED

PASCAL IS BECOMING VERY POPULAR

FORTRAN
COBOL
} DO EXIST

FORTH

C

BUSINESS APPLICATIONS USE MAINLY READY-MADE PACKAGES (e.g. VISICALC, CONTROLLER, CASHIER)

POPULARITY OF PASCAL IS DUE TO THE

UCSD PASCAL SYSTEM.

IT IS MORE THAN A COMPILER ALONE:
COMPLETE PROGRAM DEVELOPMENT AND EXECUTION ENVIRONMENT FOR SMALL COMPUTERS.

- TEXT EDITORS
- FILE MANAGEMENT UTILITIES
- COMPILERS (PASCAL, BASIC)
- ASSEMBLER
- LINKAGE EDITOR

HIGHLY TRANSPORTABLE SYSTEM (WORKS ON ≈20 MINIS AND MICROS).
BASIS IS THE UCSD p-MACHINE: SIMPLE STACK COMPUTER, IMPLEMENTED IN HARDWARE OR IN FORM OF AN INTERPRETER.

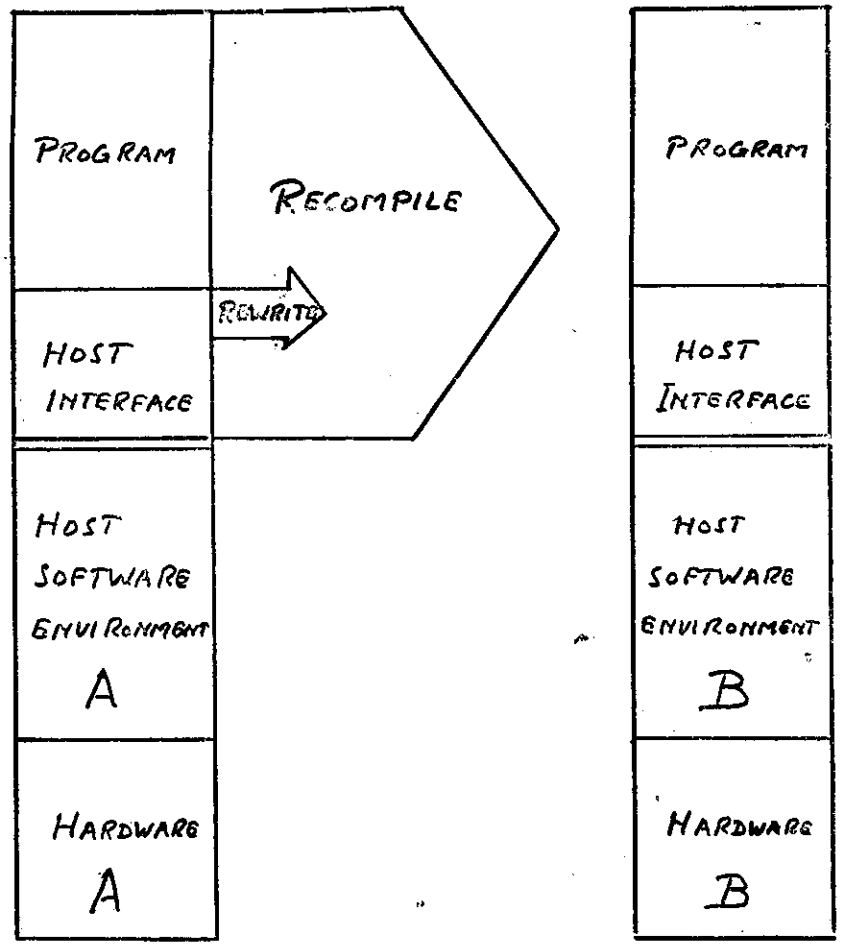
EXAMPLE OF BASIC

```
10  REM   SIMULATION OF BOUNCING BALL
20  DIM  X(100), Y(100), Z(100), T(100)
30  PRINT "INITIAL HEIGHT OF BALL (FT)";
40  INPUT H
50  IF H=0 THEN 810
    PRINT H



160 REM  INITIALIZE PARAMETERS
170
180 LET B=T(1)=X(1)=Z(1)=0
190 LET Y(1)=H


240 FOR I=1 TO 99
250     GOSUB 730
260     IF Y(I+1)>0 THEN 330
270     IF B=N THEN 360
280     LET D1=D*Y(I)/(Y(I)-Y(I+1))
290     LET Z1=-C*(Z(I)-G*D1)

330 NEXT I
340 GO TO 400


730 REM  SUBROUTINE TO CALCULATE .....

760 LET T(I+1)=T(I)+D
770 LET X(I+1)=X(I)+V*D
800 RETURN
810 END
```

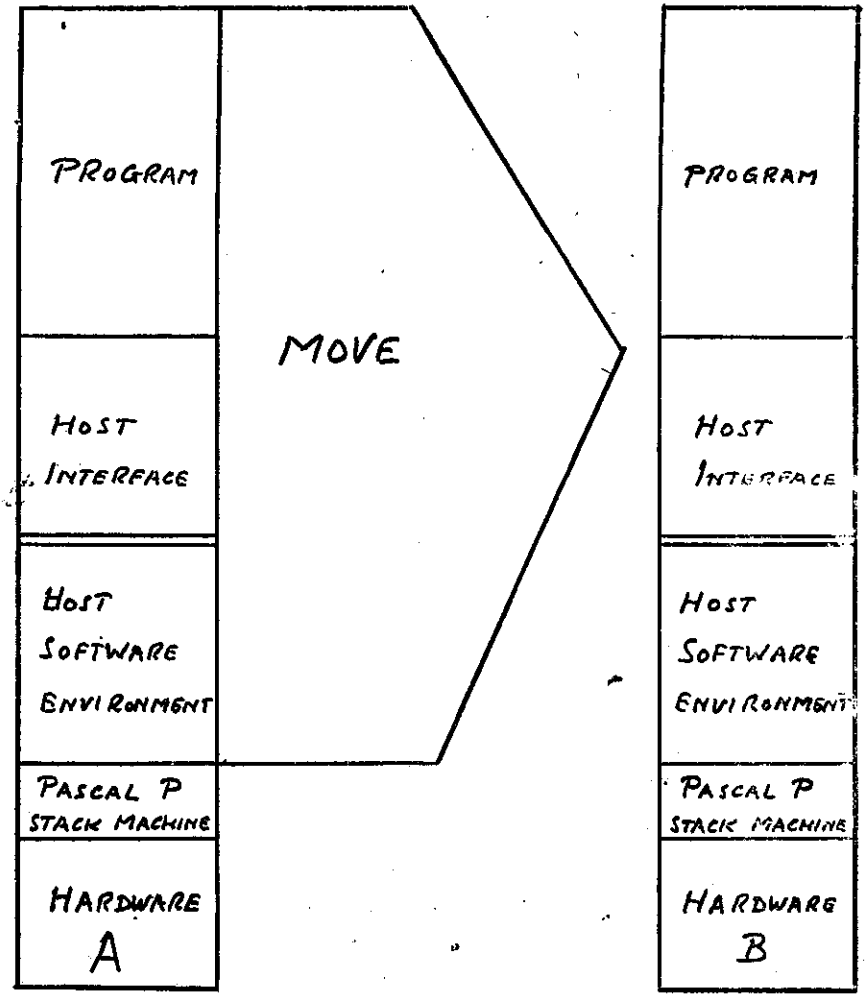

EXAMPLE OF PASCAL

```
VAR A, B, C, D, X1, X2 : REAL ; F: CHAR;
BEGIN
  D:= B*B-4*A*C
  IF D>=0 THEN
    BEGIN
      F:='R'; X1:=(-B-SQRT(D))/(2*A);
              X2:=(-B+SQRT(D))/(2*A)
    END
  ELSE
    BEGIN
      F:='C'; X1:=-B/(2*A);
              X2:=SQRT(-D)/(2*A)
    END;
END.
```

TRANSPORTABILITY



LANGUAGE APPROACH



ENVIRONMENT APPROACH

Operating Systems

Personal Computers have stimulated the development of User Friendly operating systems, running on small configurations.

Purpose of an operating system:

- Provide facilities, while at the same time:
- Hide low-level details.

Roughly two types of OS:

- General Purpose
- Real-Time (Process Control)

The second admits user programs (tasks), initiated by external interrupts.

Tasks of an Operating System:

1. Processor Management
   - essential for multiprogramming system
   - Multi-Task (I/O and Processing!)

   Interprocess synchronisation: Semaphores, Read-Modify-Write as indivisable operation.
2. Memory Management.
   - Memory Allocation
   - Overlays (Parts of same program in same space)
   - Chaining (A program can call next one from disk)
   - Segmentation (Program is cut in pieces by OS)

3. Peripheral Management.
   - Device allocation
   - Interrupt Handling
   - Device independent User-interfaces
   - Device drivers
   - Spooling
4. File Management.
   - File Allocation
   - Protection
   - Directory / Catalog
   - Drivers

Not all of these facilities are needed for a single user system.

Operating Systems for Personal Computers:

(Ref.: Electronic Design News (EDN), Nov. 5, 1980)

- UCSD-Pascal and Apple Pascal
  - Multitasking, overlays, segmentation, named files
  - Pascal + Assembler
- CP/M 2.2 (Digital Research)
  - Single User, device independence, spooling
  - Assemblers 8080, Z80
  - APL, Basic, Forth, Lisp interpreters
  - Basic, Fortran, C, Pascal, PL/1, Cobol compilers
  - Xassemblers for 1802, 8086, 6502, 6800, 8048
- CP/Net 1.0 and MP/M 1.1

  as CP/M +:
  - multitasking, multi-user.
- VersaDos for Motorola 68000
- DOS 3.3 for Apple
  - single user
  - 6502 assembler
  - Floating-Point and Integer Basic interpreters.
- DOS 68 or 69
  - 6800 and 6809 Macro-Assemblers
  - Basic and UCSD-Pascal interpreters
  - Basic, Fortran and Cobol compilers.

Several Operating Systems have been derived from Unix.
Unix was developed at Bell Laboratories for PDP-11 and set a standard for operating systems.

- XENIX
  - System vs. User modes, Multi-tasking, Multi-user
  - Assemblers for 8086, Z8000, 68000, PDP-11
  - Basic interpreter
  - C Compiler

  XENIX is a level 7 Unix operating system
- CROMIX
  - System vs User modes, Multi-Tasking, Multiuser, Multiprocessors (?)
  - Z80 macro-assembler
  - Basic and Lisp interpreters
  - Fortran, Cobol compilers.

WE GAINED A LOT IN USER CONVENIENCE WITH AN OPERATING SYSTEM.

AS THE OS HIDES THE LOW LEVEL DETAILS, WE CANNOT DO OUR SPECIAL THING WHICH NEEDS THOSE LOW LEVELS.

SUPPOSE WE WANT TO USE A HOBBY COMPUTER FOR OUR EXPERIMENT AND THAT WE WANT TO MAKE USE OF THE "HIGH LEVEL" LANGUAGE FACILITIES OFFERED.

THEN WE MUST PROVIDE TWO INTERFACES:

- BETWEEN COMPUTER AND THE HARDWARE OF THE EXPERIMENT
- BETWEEN THE HIGH-LEVEL PROGRAM AND THE LOW-LEVEL INTERFACE DRIVERS

FIRST OF ALL, MAKE SURE THE COMPUTER HAS THE NECESSARY EXPANSION CAPABILITIES.

COMMODORE PET HAS AN IEEE-488 INTERFACE (OTHER NAMES: HP-BUS, GPIB).

- CONTROL OF MEASURING INSTRUMENTS
- READING OF MEASUREMENTS

MANY MODERN INSTRUMENTS HAVE IEEE-488 INTERFACES. FOR STANDARD SEE: ANSI/IEEE Std 488-1978

APPLE HAS SLOTS AVAILABLE FOR USER INTERFACES. THE SLOTS ARE ADDRESSABLE WITH LOGICAL ADDRESSES.

MANY SPECIAL INTERFACE BOARDS ARE OFFERED FOR THE APPLE.
SEE ADVERTISEMENTS IN:

- BYTE McGRAW HILL $ 19/YR IN USA
  $ 35/YR SURFACE
  $ 43/YR AIR-EUROPE
- CREATIVE COMPUTING
- EDN
- ELECTRONICS McGRAW HILL
- ELECTRONIC DESIGN HAYDEN?
- COMPUTER DESIGN
- MICRO (IEEE COMPUTER SOCIETY)
  $ 8.00 + MEMBERSHIP DUES
  $ 22 FOR NON-MEMBERS (QUARTERLY)
- COMPUTER (IEEE COMPUTER SOCIETY)

A CAMAC INTERFACE HAS BEEN DEVELOPED FOR THE APPLE (CAMAPPLE) AND IS DESCRIBED IN: PROC. TOPICAL CONF. ON APPL.[d] MICROPROC.[s] IN HEP EXPERIMENTS, GENEVA, MAY 1981, CERN 81-07 (OKOBY ET.AL. pp 294-302)

CAMAPPLE CONSISTS OF TWO CIRCUITS:

- GENERAL PURPOSE I/O SUBSYSTEM, HOUSED IN THE APPLE. BASED ON 6522 VIA (VERSATILE INTERFACE ADAPTOR)
- CAMAC INTERFACE, HOUSED IN A CAMAC CRATE.

CNAF IS TRANSFERED IN 4 CYCLES, VIA THE B-PORT OF THE 6522. THE LEAST SIGNIFICANT BITS OF THE B OUTPUT IDENTIFY THE INFORMATION:

00	:	5 BITS F (B2-B6)
01	:	2 BITS C (B2-B4)
10	:	5 BITS N (B2-B6)
11	:	4 BITS A (B2-B5)

B1
B0

B7 STARTS A CAMAC CYCLE.

DATA IS TRANSFERRED ON THE A PORT DATA DIRECTION REGISTERS MUST BE SET BY SOFTWARE IN AGREEMENT WITH THE FUNCTION (F) PERFORMED.

F = 0-7 IS READ FROM CAMAC
F = 16-23 IS WRITE TO CAMAC.

AN I/O SUBSYSTEM ON APPLE MAY CONTAIN 256 BYTES OF MEMORY, ACCESSED BY EXECUTING PR#n (n IS SLOT NUMBER)

A PROM IN CAMAPPLE CONTAINS THE NECESSARY PROGRAM, WHICH HAS ACCESS TO THE BASIC VARIABLES F,C,N,A,D AND IR.

THE SEQUENCE

```
F=0
C=2
N=15
A=4
PR#3
PRINT D
```

WILL PRINT CONTENTS OF SUBADDRESS 4 AT STATION 15 IN CRATE 2 (LEAST SIGNIFICANT BYTE, 24 BITS CAN BE READ BY REPEATING PR#3 PRINT D 3 TIMES.

MANY VERSIONS OF BASIC CONTAINS COMMANDS:

- PEEK :

```
LET A = PEEK (255)
```

STORES CONTENTS OF ABSOLUTE MEMORY LOCATION 255 IN VARIABLE A.

- POKE:

```
POKE (255, 10)
```

STORES 10_{10} IN ABSOLUTE ADDRESS 255

THIS ALLOWS TO EXCHANGE ONE BYTE AT THE TIME WITH THE EXTERNAL WORLD, VIA A PIA FOR INSTANCE.

- SPEED: HUNDREDS OF MICROSECONDS / BYTE.

PROBLEM:

CAN WE TRANSMIT A BLOCK OF DATA FROM AN EXPERIMENT (WITH DMA FOR INSTANCE) AND THEN ACCESS THE DATA FROM A HIGH-LEVEL LANGUAGE AS IF IT WERE AN ARRAY?

THE PROBLEM ARISES BECAUSE THE OS STORES VARIABLES WHERE IT THINKS FIT, NOT WHERE WE WANT THEM TO BE.

A DMA CONTROLLER THEREFORE DOES NOT KNOW WHERE TO STORE THE DATA.

CASE 1:

THE HIGH-LEVEL LANGUAGE AND OS (LOADER) PROVIDE LINKAGE TO ROUTINES WRITTEN IN ASSEMBLER (e.g. UCSD PASCAL)

THE FOLLOWING ROUTINE WILL THEN BE ABLE TO TRANSMIT THE STARTING ADDRESS TO A DMA CONTROLLER:

```
        XDEF    BUFFER
SETDMA  LDAA    BUFFER      AT EXECUTION
        LDX     SETDMA+1    TIME LOCATION
        STX     DMACON      SETDMA+1 AND
                            SETDMA+2 WILL
                            CONTAIN THE
                            ABSOLUTE ADDRESS
                            OF BUFFER
        :
        :
DMACON  EQU     CBA0        ADDRESS OF DMA
                            CONTROLLER.
BUFFER  BS2     500
```


CASE 2:

THE "HIGH-LEVEL" LANGUAGE DOES NOT PROVIDE LINKAGE TO ROUTINES WRITTEN IN ASSEMBLER (e.g. BASIC INTERPRETERS).

THE SYMBOLIC NAMES FOR BASIC VARIABLES REMAIN THEN UNKNOWN TO A PROGRAM WRITTEN IN ASSEMBLER.

ACCESS TO A VARIABLE ARRAY DECLARED IN BASIC CAN THEN BE OBTAINED FROM A USER PROGRAM WITH THE HELP OF VARPTR.

AT EXECUTION OF BASIC STATEMENT

$$X = USR(AZ\%(0))$$

THE VARIABLE POINTER VARPTR CONTAINS THE ADDRESS OF THE FIRST ELEMENT OF THE ARRAY AZ%

KNOWLEDGE OF THE ABSOLUTE ADDRESS OF VARPTR (e.g. 0096-97 FOR PET, 0083-84 FOR APPLE) ALLOWS THE USER PROGRAM TO ACCESS THE ARRAY. DATA MAY THEN BE READ FROM THE OUTSIDE WORLD AND STORED IN THE ARRAY FOR PROCESSING BY A BASIC PROGRAM.

WHERE TO PUT IN MEMORY THE USER PROGRAM?

- TYPE IT IN AS A REM IN THE BASIC PROGRAM
- PUT IT IN THE ARRAY, TYPE IT IN AS DATA
- STORE IT ABOVE HIMEM.

DETAILS IN: NGUYEN NGOC HOAN, LINKING BASIC VARIABLES TO BINARY DATA WORLD

CONCLUSION:

YOU CAN USE A PERSONAL COMPUTER FOR REASONABLY FAST TRANSFER OF DATA BLOCKS TO/FROM EXPERIMENTAL EQUIPMENT, AND THEN PROCESS THE DATA.

IT IS HOWEVER MESSY, PARTICULARLY IN THE CASE OF A BASIC INTERPRETOR.

NEAT SOLUTION: THE BASIC INTERPRETER ITSELF SHOULD RECOGNIZE COMMANDS ALLOWING REASONABLE COMMUNICATION WITH EXTERNAL, NON-STANDARD DEVICES.

THIS IS DONE IN CAVIAR.

REFERENCES:

- DATAPRO REPORT ON PERSONAL COMPUTERS (JUNE 1980)
- JACK HEMENWAY, μC OPERATING SYSTEMS DIRECTORY, EDN NOVEMBER 5, 1980.
- MARK OVERGAARD, UCSD PASCAL: A PORTABLE SOFTWARE ENVIRONMENT FOR SMALL COMPUTERS, PROC. NAT. COMPUTER CONF. 1980, pp 747-754

EVOLUTION: STAND-ALONE
BATCH
TIME SHARING
↓
PERSONAL COMPUTER

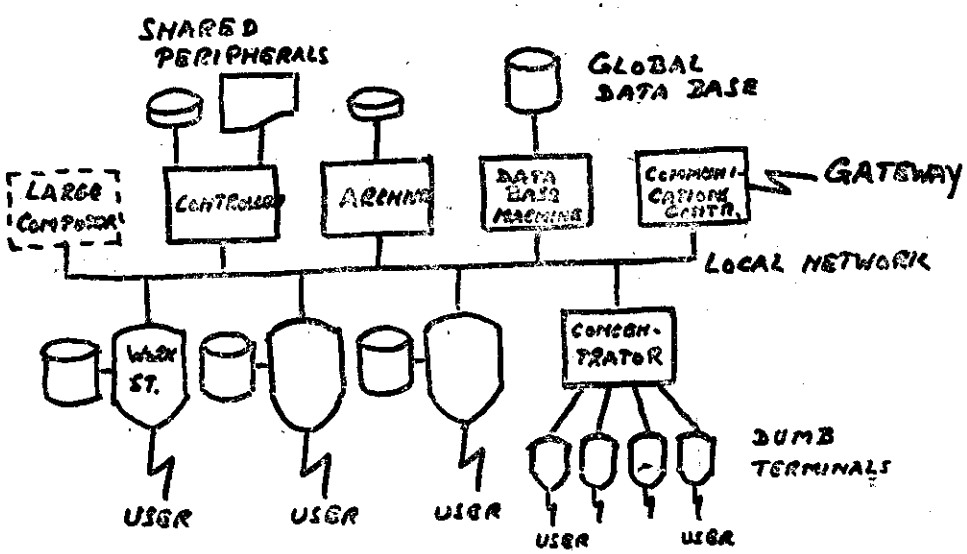
WEAKNESSES OF TIME-SHARING SYSTEMS:

- ECONOMIC: PURCHASE + RUNNING COSTS HIGH.
- OPERATIONAL:
  - INDIVIDUAL PRODUCTIVITY AFFECTED BY LOAD ON THE SYSTEM.
  - HARDWARE FAILURES AFFECT ALL USERS
  - DIFFICULT TO EXPERIMENT WITH NEW SYSTEM SOFTWARE
  - DIFFICULT TO EXPERIMENT WITH HARDWARE
- TECHNICAL: NOT POSSIBLE TO ACHIEVE HIGH BANDWIDTH INPUT/OUTPUT WITH SEVERAL USERS (VIDEO OUTPUT, NON-KEYBOARD COMMUNICATIONS).

TREND TOWARDS POWERFUL PERSONAL COMPUTERS, WITH
- VIDEO DISPLAY
- TABLET
- LIGHT PEN
- FLOPPY DISK
- MINI HARD DISK

ETC.

THESE "WORKSTATIONS" ARE LINKED TOGETHER AND TO A "DATA BASE MACHINE" VIA LOCAL AREA NETWORK



SOME SYSTEMS IN USE:

+ ALTO AT XEROX PARC
  ETHERNET
  ALTO IS A POWERFUL MACHINE, BUILT FROM DISCRETE COMPONENTS. PRODUCED IN LIMITED QUANTITY FOR XEROX INTERNAL USE.

- PERQ (THREE RIVERS COMPANY) WAS (IS?)
  COMMERCIALLY AVAILABLE

- APOLLO DOMAIN.
  DESIGNED BY PEOPLE FROM 3 RIVERS.
  BASED ON 2 MOTOROLA 68000's:
  - ONE FOR PROCESSING
  - ONE FOR DISPLAY

- HIGH RESOLUTION RASTER SCAN GRAPHICS,
- FAST ETHERNET- OR CAMBRIDGE RING-LIKE LOCAL AREA NETWORK
- SHARING OF EXPENSIVE RESOURCES (DISCS, PRINTERS)

ARE ESSENTIAL INGREDIENTS OF SYSTEMS BASED ON PERSONAL COMPUTERS FOR PROFESSIONALS.

LILITH!