

# *in* STRIDE

October 1988 - Vol. 1, Issue 2

## CONTENTS:

Updated! Updated!

CMOS Memory Map

Four Quick Ways to  
Interrogate Stride Hardware

APL68000

WORD? File Extension

Disk Space

New Tape Backup  
Program and Update

Yog's California  
Earthquake Hot Line

ASE File Lookup

# tech notes

## Updates! Updates!

This issue attempts to bring you up-to-date on some of the changes that have occurred to the more well-known products on Tride's machines. Besides letting you know what they are, Test Notes is making it easy to get the updates via a special order form in the middle of this issue. Most of the updates discussed can be purchased at a small cost (most are \$25) which covers shipping and handling, etc.

Not all of the updates are new, but you may have missed the announcement or felt that you were going to wait awhile before getting updated. Tride has also changed the update policy slightly this year. Hopefully we can make changes available to you sooner.

Most computer products, software and hardware, are constantly changing

and growing. Some computer manufacturers release every four to six months which incorporate all the small changes that occur that period. If Tride is a distributor of a product handled this way by its vendor, we can only update you when the vendor updates us. Most vendors also require a new receipt for the new version. This is the case with the p-System IV.25 update from version IV.20.

However, products developed internally by Tride Microsystems, such as our P800 and 80-P800 are handled a bit differently. As time changes its face, an update diskette is created and given to Tech Support. If a user calls in wanting that change, he can get the update diskette immediately for only \$25!

Tride does not automatically update every user, but, through the Test Notes, we will be letting our registered users know about the changes.

What is a registered user? If you're receiving Test Notes you're registered. Any Tride/Trage owner can register simply by sending in their name,

address and machine serial number to Tride Microsystems at our Exam address. Just having a machine does not automatically register you since most machines and software are sold through dealers and distributors -- Tride Microsystems does not know who you are. There is a card in the shipping box that new owners are supposed to fill out and return. You would be surprised how many don't!

Updates are announced through its Tride, and our reader bulletin. Major announcements also are made over the various electronic mail systems.

Updates on certain products require proof that you purchased the original product. Your Purchase Order number or the Serial number of products (such as CP/86) and the Serial Number of your Tride/Trage machine are necessary in order to purchase the update.

If an item on the order form in the back is not clear to you, look through this issue. Most items are explained with a short paragraph. The p-System update was explained in detail in the last issue of Test Notes, Volume 1, ☐

## CMOS Memory Map

The CMOS Memory Map shows on page 219 of the February Tride Owner's Manual's incomplete (refer to page 140 of the September edition). The CMOS memory map is shown correctly on the right.

Note that there is a 1600 byte area (FFA000 - FFAFFF) reserved for applications programs. As the battery backup circuit protects all CMOS, this area is handy as a small RAM disk for useful information. CMOS RAM is a bit slower than regular RAM -- but a bit faster than disk.

Other applications are allowed to use this area, or it is best to initialize it at the start of your program. If used to transfer data between programs, make sure that there is no way another program can be called in between that might kill the area. Most operating systems have ways that programs can chain or pipe to each other safely.

These locations are specific to the 486 series, the Sage computers do not have an equivalent CMOS area.

### 486 (or 486) CMOS MEMORY MAP

FFFF00 - FFFFFF	000	Reserved for BIOS expansion - DON'T change!
FFFF00 - FFFFFF	01	BIOS Update & Backward Comp
F00000 - F0000F	50	Shared between the ROM port 0000 to the disk and the ROM port 0000 to the PROM boot loading.
F00000 - F0000F	52	INT 0000-0004
F00000 - F0000F	56	INT 0004
F00000 - F0000F	5C	CMOS Real-time Device Data Area Can only be updated with the disk flag Reserved for BIOS
F00000 - F0000F	6000	Reserved for user applications. ***** THIS AREA
F00000 - F0000F	1000	Reserved for BIOS expansion. Do not manually update as it is of 2k x 2 New page 0000 to 0000 by default (page 0000) Old page 0000 0000 0000 0000 (page 0000) 00 to 0000 of 0000 0000 to 0000 is reserved on page 000 This area appears to be there as it is "changed" with the variable flag -- in context of page.
F00000		Start of 486 system BIOS
Also see table in the Manual's Appendix for these reserved memory locations.		
F00000 - F0000F		Reserved for Tride BIOS
F00000 - F0000F		Reserved for Tride BIOS

## Four Quick Ways To Interrogate Stride Hardware.

When a program must run on different kinds of hardware, it is very useful for the program to know how much memory it has, what version of the system is running, etc. The following routines are part of the STRIDECODE and BASICCODE programs released with the IV-30 SYSTEM, but can also be converted to almost any operating system that allows access to memory or BIOS calls. (PDSB doesn't.)

### Which Machine?

The first program is a simple procedure that tells you which machine, 486 Turbo or Sage, that you have by a BIOS call. On the Stride 486 machines, the string "STRIDE" will be returned. The characters returned from a BIOS may vary and may not be printable characters.

### How Much RAM Memory?

The second program determines the amount of RAM memory in the machine. Change ADDRESS to ADDRESS in the line indicated to read the memory size of a Sage machine.

### What Version Of The BIOS?

The third program works for either a BASIC or 486 Stride machine and determines what version of the BIOS or BIOSCODE you have. This can be used to tell your program if Cache is available to you or not since various prices for E-3 did not have Cache. (It doesn't tell you if Cache is turned on.)

### How Big Is The Disk?

The fourth program reads the hard disk drive map area to determine how large a disk you have. See the comments to change it to run on Sage machines. By changing the variable WDEV from 0-3, you can do this for all four hard disk drives. □

```
PROGRAM WhichMachine;
VAR Data:STRING;
    I:INTEGER;
BEGIN
  Get;
  FOR I:=0 TO 255 DO
    Data:=Data+CHR(I);
  Data:=Data+CHR(0);
  IF Data[0]#STRIDE THEN
    WriteLn('No Stride machine found');
  IF Data[0]#SAGE THEN
    WriteLn('No Sage machine found');
END;
```

The program above checks for a Sage or 486 Stride machine.

```
PROGRAM MemorySize;
CONST MaxMem=1024;
VAR MemSize:INTEGER;
    I:INTEGER;
PROCEDURE GetMemory;
VAR Address:INTEGER;
    Data:STRING;
BEGIN
  Address:=0;
  WHILE Address<MaxMem DO
    Data:=Data+CHR(Address);
    Address:=Address+1;
  END;
  MemSize:=Length(Data);
END;
```

```
VAR MemSize:INTEGER;
    Address:INTEGER;
    Data:STRING;
    I:INTEGER;
    MemSize:=0;
    Address:=0;
    WHILE Address<MaxMem DO
      Data:=Data+CHR(Address);
      Address:=Address+1;
    END;
    MemSize:=Length(Data);
    WriteLn('Memory Size: ', MemSize, ' bytes');
END;
```

The program above finds the memory size of the machine.

```
PROGRAM BIOSVersion;
VAR BIOS:STRING;
    I:INTEGER;
    BIOS:=CHR(0);
    FOR I:=0 TO 255 DO
      BIOS:=BIOS+CHR(I);
    END;
    BIOS:=BIOS+CHR(0);
    IF BIOS[0]#STRIDE THEN
      WriteLn('No Stride machine found');
    IF BIOS[0]#SAGE THEN
      WriteLn('No Sage machine found');
    END;
```

The program above reads the BIOS version number.

```
PROGRAM Section)

```

```
DATE [01/01/79, 0000.0000] Author: J. Cunningham
```

```
{
  For a RAM Disk, change the DR0 (low address)
  Address: 00000000:0000:0000, with: 00000000:0000:0000
  Address: Configuration: in the Configuration.
}

```

```
RAM (RAMDISK) (RAM);
RAM (RAMDISK)
  RamDisk: Configuration: [RAMDISK] (RAMDISK)
  RamDisk: Configuration: [RAMDISK] (RAMDISK)

```

```
RAMDISK: Configuration: [RAMDISK] (RAMDISK)
RAM
RAM: (RAM);
RAM: Configuration: [RAMDISK] (RAMDISK)

```

```
RAM
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
{
  * Loop through the address that the DR0s which need to set
  correctly, the disk address is set right. The loop below
  is set up so the DR0s which need to be set to zero
  is loop.
}

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
{

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

```
RAM: Configuration: [RAMDISK] (RAMDISK);

```

## Running With Both CACHE & RAM Disk

In last month's Tech Notes, we discussed the new CACHE feature of the 486 Series BIOS (page 8 of that issue). Since then, we've had some feedback on the operation of the feature. One user reported that his application, generating a report, ran 3 times faster under CACHE operation.

While originally intended that CACHE be a substitute for RAM Disk, it is better protected and does not require loading any files. However, during this test it was found that most applications were designed around RAM Disk and needed that area.

Additional changes were made to the BIOS to allow these users. The final release of the BIOS, (DRS, DRD, DRB) will allow both CACHE and RAM Disk area in the Multi user BIOS. (That will be the single user). This was fairly easy to implement as MULTIBIOS already supported both RAM Disk. The first RAM Disk became the CACHE Disk.

Both CACHE and RAM Disk are memory lags. It is not recommended that a CACHE be much smaller than 256K bytes. Disk performance may actually depend if too small of an area is assigned.

### Installation

The MULTIBIOS and not RAM Disk #0 address (translator and enable) or enable. Set RAM Disk #1's low address to the high address of RAM Disk #0 (CACHE). Enable installation.

Use MULTIBIOS to define which user gets access to the RAM Disk #0. All users get the advantages of CACHE. In the channel map of each user, map #0 to #00 (RAM Disk #0) or device 000.

Save the new Multibios configuration and reboot.

### Operation

CACHE will be available to the operating system. RAM Disk #0 will appear in the p-System File as the first RAM Disk, RAMDISK0 [ ]

The program above reads the hard disk size.

## APL 68000:

More Than A Language, It's  
A Problem Solving Tool

By Sabers Software, Ltd.

APL is considered a THILL (Very High Level Language). It employs a High level of abstraction, isolating the user from machine activities, thus freeing him to concentrate on the problem he wants to solve. In that respect, APL is a human oriented language, not a computer oriented one. APL was a very brave system, representing basic functions by symbols rather than keywords. APL features a unique character set, thus requiring a special APL/ASCII terminal.

The use of special symbols is one of the reasons why the language has been debated whether from experts to dilettos and hard to learn.

Actually, the contrary is true. APL is a language that is easily learned. Its terseness and compactness make it very powerful and flexible and can increase development productivity significantly.

Take a minute and think about how much time you would save if the language you are currently using had these APL features: data types that did not require declaration in advance, operators that maximum error did not arise that can be operated on without the need for loops.

IBM internal statistics have shown that by using APL, a 1-10 fold decrease in development time can be achieved.

### History of APL

APL was originally conceived as a mathematical notation by Kenneth Iverson at Harvard, an effectively describe algorithms in applied mathematics to his students.

APL shows its mathematical roots in the terminology applied to the language (rank for the dimension of an array, scalar, inner and outer product) and also in its heavy use of mathematical symbols and expressions.

After Iverson joined IBM in 1960, a development phase began that was followed by the first actual computer implementation of the language in 1968 on an IBM/360.

```

# LET'S START WITH SIMPLE ARITHMETIC.
# TIME # INVOKED A COMMENT LEVEL
#++

1
1.2345

11.44#
# POSITIVE OPERATORS EXTEND TO VECTORS & ARRAYS
# REDUCTION IS FROM RIGHT TO LEFT, NO EXCEPTIONS

D←TABLE←% 10000 # CREATE A MATRIX (D-D ARRAY)

88 378 280 260 110 24 840 240 688 180
280 410 18 27 288 308 4 192 24 200
648 885 455 404 264 48 827 308 381 488
282 182 24 384 188 877 876 408 188 184
482 282 877 328 27 216 448 187 876 884

D←TABLE # AND MULTIPLY THE ABOVE TABLE BY 3

108 1134 890 864 330 72 1020 1020 1004 876
780 1248 54 81 792 1008 12 876 102 827
1080 885 1098 1078 792 188 980 104 1002 1288
1164 288 72 1187 888 864 1187 1488 548 878
1478 1088 1128 878 112 888 1028 811 887 1558

# AND DISPLAY OF ALL ABOVE BY THE HIGHEST COLUMN.
TABLE←J TABLE

88 378 280 260 110 24 240 240 688 180 2448
280 410 18 27 288 308 4 192 24 200 1784
384 885 455 404 264 48 827 308 381 488 3181
282 182 24 384 188 877 876 408 188 184 2571
482 282 877 328 27 216 448 187 876 884 3648

# CREATE A VECTOR OF NUMBERS FROM 1-10 AND ADD 10.
D←10J+10 10 10

11 12 13 14 15 16 17 18 19 20

+JLIST # SHOW IT ALL UP

111
JLIST # SHOW MANY ELEMENTS IN LIST

12
(+JLIST)←JLIST # AND MAKE IN THE MEAN

12.1
# MULTIPLY THE LIST BY 10 (RANDOMLY CHOSEN)
LIST←10J*JLIST
JLIST # JUST DISPLAY THE NEW CONTENTS

12 18 24 30 36 42 48 54 60 66

17 24 31 38 45 52 59 66 73 80
JLIST # AND ADD THE LARGEST NUMBER

188
# WE CAN MANIPULATE CHARACTERS THE SAME WAY
D←+STRING←D J
A←INDEX←CHARACTERS←INDEX←JLIST
JSTRING # SHOW MANY ELEMENTS IN STRING

18
J STRING # REVERSE OUR VECTOR
INDEX←JLIST
INDEX←JLIST
STRING←J # ARRAYS CAN INDEX ARRAYS OR VECTORS

488
J STRING # PICK ALL ELEMENTS BUT THE FIRST ONE
A←INDEX←JLIST
```

Between 1963 and 1966, the language syntax was simplified and its functions were extended. The definition of the character set was based on the IBM 3600 standard, which utilized the relative point elements. The limitations of this 48 element character set led to two important syntax definitions to represent both monadic (one argument) or dyadic (two arguments) by the same symbol and to generate computer symbols by means of overloading built-in characters.

Because APL evolved and evolved in an academic environment, its notation and built-in design philosophy was not initially restricted by the limitations of real-world computer systems, as is the case with most other programming languages.

This is one reason APL should not be thought of as a language, but as a highly sophisticated set of tools for computer processing.

#### APL-36000 and the STRIDE

APL-36000 is a superset of the IBM 360 APL six-level standard implementation. It is available on SAUCE and STRIDE computers under the MIRAGE multiplex, time-sharing operating system.

Unlike many other multiplex, multi-tasking operating systems, MIRAGE has many features that make it an ideal environment for APL. APL-36000 and the MicroAPL utility libraries allow the user to completely control a MIRAGE environment from within APL. There are even functions that allow it to temporarily leave APL, call to MIRAGE, perform certain functions or run a program, and return to the APL workplace.

APL-36000 offers a sophisticated multiplex file system, access to the MIRAGE printer operator, and interfaces to 36000 Assembly and other language compilers available under MIRAGE. The computer file system treats files as a collection of records, each one of arbitrary length or type. In addition, a filing utility for keypad record access is available.

#### APL Calculator Mode

To get a feeling for APL-36000, look at the examples on the previous page. It shows several expressions for operations commonly needed in programming and how to implement them in APL. Array manipulations are easily done in APL with simple one-line commands.

## APL FUNCTIONS

### THE STANDARD FUNCTION DEFINITIONS

```
Ⓐ SAMPLE          ARRANGE A VECTOR WITH ALL ARRANGEMENTS
                   AND THE RESULT SORTED IN ASCENDING
Ⓐ SAMPLE Ⓜ       J+Ⓜ (ASCENDING SORT) (ASCENDING SORT)
                   (ASCENDING SORT)
Ⓐ SAMPLE Ⓜ       J+Ⓜ (ASCENDING SORT) (ASCENDING SORT) (ASCENDING SORT)
                   (ASCENDING SORT)
Ⓐ SAMPLE Ⓜ       J+Ⓜ (ASCENDING SORT) (ASCENDING SORT) (ASCENDING SORT)
                   (ASCENDING SORT)
Ⓐ SAMPLE Ⓜ       J+Ⓜ (ASCENDING SORT) (ASCENDING SORT) (ASCENDING SORT)
                   (ASCENDING SORT)
Ⓜ
```

```
Ⓐ APLICE
Ⓐ APL J+Ⓜ
Ⓜ Ⓜ A SAMPLE FUNCTION TO CALCULATE A FEW
Ⓜ STATISTICS ON A VECTOR OF NUMBERS
Ⓜ (1) N←N/Ⓜ/VECTOR
Ⓜ NUMBER OF MEMBERS: ⓂN
Ⓜ EXPANDED: ⓂN | Ⓜ | ⓂN
Ⓜ MEAN: Ⓜ/Ⓜ+Ⓜ/Ⓜ+Ⓜ
Ⓜ VARIANCE: ⓂN*(Ⓜ/Ⓜ+Ⓜ/Ⓜ+Ⓜ-Ⓜ)/Ⓜ+Ⓜ+Ⓜ
Ⓜ STANDARD DEVIATION: Ⓜ/Ⓜ+Ⓜ+Ⓜ
Ⓜ-END
Ⓜ APLICE 'MOVEMENT MUST BE IN APLICE' + ⓂN
Ⓜ AND END OF PROGRAM', ⓂN
Ⓜ
```

```
Ⓐ APLICE
Ⓐ APL
Ⓜ (1) ⓂN IN ONE STATEMENT OF GENERATING PRIME NUMBERS
Ⓜ (2) IN APL, IT USES JOCKING (SEARCH IN ORDER) NOT
Ⓜ FORK (NO ADDITIONAL STORAGE FOR COMPUTATION)
Ⓜ (3) THERE IS A SIMPLE WAY TO GENERATE PRIME
Ⓜ NUMBERS WITHOUT JOCKING
Ⓜ J+Ⓜ + Ⓜ+Ⓜ(ⓂN) | Ⓜ
Ⓜ Ⓜ(Ⓜ+Ⓜ+Ⓜ | Ⓜ(Ⓜ-Ⓜ) | Ⓜ+Ⓜ+Ⓜ+Ⓜ) | Ⓜ
Ⓜ N←Ⓜ | Ⓜ+Ⓜ+Ⓜ
Ⓜ Ⓜ(Ⓜ | Ⓜ+Ⓜ+Ⓜ) | Ⓜ PRIME FOUND
Ⓜ
```

## APL Functions

In fact, the statements in the example look much like a sophisticated calculator's statements. Correct? Up to now, we have only used APL in its first calculator mode. The real power of APL is the ability to define functions that behave exactly the same way the built-in functions do.

The same rules that apply to primitive operators, right-to-left order of evaluation, normal (two argument) or dyadic (two arguments), apply also to user-defined functions. In addition, a user-defined function can be infinite (no arguments) and can explicitly return a result or no result.

The syntax for function definitions is shown in the box on page 7.

The second box shows a few examples of some simple functions using the APL syntax.

User-defined functions cannot be saved on an element-by-element basis. In APL, a program is a collection of independent user-defined functions that can be kept in libraries and copied (COPY command) into a workspace. Many functions have 10 or fewer lines and are easy to write and maintain.

### APL SYSTEM Commands

APL is organized around the concept of a **WORKSPACE** (with the similarity to today's popular desktop computers). The interpreter offers a large set of system commands to manage your workspace.

All system commands begin with a left parenthesis "(" which must appear in the first column of a valid APL statement.

To the right, you will see a list of some of the system commands. In addition to SYSTEM commands, a variety of SYSTEM functions and variables provide the user with built-in utilities.

(Continued on page 8)

## APL SYSTEM COMMANDS

[AV	ATOMIC VECTOR
[A	APPARENT ARGUMENTS
[AI	ACCOUNT INFORMATION
[AS	ACCOUNT NUMBER
[C	CPU TIME (REAL)
[CZ	CONSTANT TIME
[D	KEYBOARD UNLOCK TIME
[E	SEARCHABLE ACCESS UNIT
[F	CONTROL CHARACTER
[FC	TERMINAL INPUT/OUTPUT
[G	CONSOLE CONTROL
[FMT	OBJECT'S FORMATTING

## APL SYSTEM

## WORKSPACE COMMANDS

[CLEAR	CLEAR THE ACTIVE WORKSPACE
[COPY	COPY SELECTED OBJECTS FROM STORED WS INTO THE ACTIVE WORKSPACE
[DWIDTH	SET MAXIMUM NUMBER OF SIGNIFICANT DIGITS FOR OUTPUT
[LIBRARY	REMOVE SPECIFIED WORKSPACES FROM USER LIBRARY
[LIBRARY	REMOVE GLOBAL OBJECTS FROM ACTIVE WORKSPACE, I.E. FUNCTIONS, VARIABLE GROUPS
[LIST	LIST ALL FUNCTIONS IN ACTIVE WORKSPACE
[LIB	LIST NAMES OF ALL WORKSPACES IN SPECIFIED LIBRARY
[LOAD	REPLACE CURRENT WS WITH A COPY OF A STORED WS
[PAGE	SAVE CURRENT WORKSPACE ON DISK
[PAGESIZE	SET OR DISPLAY CURRENT PAGESIZE (1-25)
[NAME	LIST NAMES OF GLOBAL OBJECTS (VARIABLES) IN WS

## Performance & Versatility

Simple system rules, many powerful primitive operations, few expensive types and unconflicated execution rules make APL a versatile multi-purpose programming language suitable not only for number-crunching applications, but also for database management, statistical analysis and office automation systems.

APL is an interpretive language, an interactive programming environment and it actually aids the user in developing problem solutions.

The interpreter runs on many, especially in Stride hardware.

Here are a few benchmark tests comparing APL/80000 on a 16 MB Stride 400 with IBM APL on an IBM-PC and STAC APL Plus on a IBM-PC.

APL/80000 is distributed in the U.S. by Network News, 2211 Old Bayshore HWY., Huntington, CA. 94022 (415)-847-0787

In Europe, contact: Sabera Software, Ltd., Unit 17, Tilbury Industrial Estate, 87 Killinghall Lane, London S20 2BP. Tele 01-471 8000.

## UNIX APL

UNIX (and also any other APL) users should take heed -- APL is being ported to Stride's UNIX and will be available in the near future.

Watch for an announcement here.

## BENCHMARK (IN MILLIBYTES)

## FUNCTION

IBM APL 80000 APL/PLUS 80000

1. FLOW REDUCTION	$Z = \sum V^2$	0.04	0.18	0.01
2. LOGICAL REDUCTION	$Z = \sum \sqrt{V}$	0.02	0.09	0.01
3. MAXIMUM REDUCTION	$Z = \sum (V) \cdot \text{AND}$	90.00	22.00	0.00
4. REPRESENTATION	$Z = \sum V^{-1}$	890.00	2400.00	2000.00
5. ABSOLUTE VALUE	$Z = \sum  V $	80.00	180.00	38.00
6. INDEXING	$Z = \sum (V)(1.00)$	20.00	14.00	7.00
7. SORTING	$Z = \sum \sqrt{ V }$	800.00	110.00	48.00
8. TAKE	$Z = \sum  V $	0.00	24.00	4.00
9. MEMBERSHIP	$Z = \sum V < 1.00$	100.00	100.00	10.00
10. TRANSPOSITION	$Z = 1 \cdot \text{SUM}$	600.00	90.00	10.00
11. OUTER PRODUCT	$Z = V^T \cdot \text{SUM}$	900.00	161.00	20.00
12. OUTER PRODUCT	$Z = V^T \cdot \text{SUM}$	8500.00	400.00	00.00
13. INNER PRODUCT	$Z = \sum (V) \cdot \sum V$	210.00	540.00	00.00
14. MATRIX DIVISION	$Z = \text{SUM} \cdot \sum V^T \cdot \sum V$	70.00	2000.00	100.00
15. MULTIPLICATION	$Z = \sum V \cdot \sum V$	100.00	400.00	00.00
16. DIVISION	$Z = \sum V \cdot \sum V$	100.00	700.00	100.00
17. LOGARITHM	$Z = \sum  V $	100.00	4000.00	1000.00
18. SINE	$Z = \sum \sin(V)$	411.00	20000.00	1000.00
19. SINE (in BENCHMARK)	(APL/PLUS/STAC)	1000	1000	004

The variables used in these benchmarks are defined as follows:

```

APL ← 16 16 16 V ← (RAND 1 0 1)/4.000
V1 ← 1 0 1 1 0 0 0 1
APL ← 10 10 10 V1N ← V1 1 0 1
APL ← 16 16 16 V1C ← ABCDEFGHIJKLMNOPQRSTUVWXYZ
    
```

All timings for IBM-APL were taken with the SORT math expression.  
(IBM-APL REQUIRED for SORT to be present).

The timings for STAC APL are without an FPS.  
APL/80000 timings are without an FPS on the Stride.



## SALE — \$150 Buys A Stride/Sage QVT102 Terminal!

The Stride Sage Factory is clearing out our used QVT102 terminals. These are working terminals, in good condition and fully functional, less than a year old. They were used in—bought by our Stride employees and are being replaced by the newer Stride terminal. A limited supply is available at \$150

each. No warranty is included, all units are on an "as is" basis.

Some have a "SPACE" key as shown in the photo, some have the QVT key. Sorry, but you don't get to choose. See section in "hand" versus "light".

One or two color QVT102's are available, please check with Stride Sage

on order availability.

No discounts apply. To order, see the form on page 11 in this issue or call Allison Brown at (700) 333-3444.

This is a great buy for a good terminal. Offer is open while supply lasts. ☐



Sealed Micro  
P.O. Box 800110, Reno, NV 89500-0010

# Update Order Form

To be eligible for some updates, you must prove prior purchase (attach a copy of the PO, invoice and/or the software serial number) and the serial number of your machine. No discounts apply. All sales are final. Materials are not included, some items will have release dates.

NAME: \_\_\_\_\_ DATE: \_\_\_\_\_  
 ADDRESS: \_\_\_\_\_ TELEPHONE: \_\_\_\_\_  
 \_\_\_\_\_ Time to call (above number): \_\_\_\_\_

Part #	Description	Price	
080000	CPU Circuit Board Packer.....	\$ 85	
080000	Workstation Board Circuit Board Packer.....	\$ 85	
080000	Used Spare 977000 Terminal.....	\$ 100	
000100	8" Sealed Universal Cable adapter kit.....	\$ 80	
All the parts needed to make an adapter between a 5000 80000 cable to a Sealed "photocopying" cable.			
080100	400 Series BIOS Update.....	\$ 85	
See BIOS, 80,8100, UTIL, 80,UTIL			
080410	Streaming Tape Partition Backup Program.....	\$ 85	
Backs up and restores hard disk partitions. Requires 800000 or p-System IV.20 update.			
SP0100	Page 12/19 Run-time Update to p-System IV.20.....	\$ 80	
SP0100	Page 12/19 Development Update to p-System IV.20.....	\$ 100	
*** Requires prior Development purchase.			
SP0104	400 Series Run-time Update to p-System IV.20.....	\$ 80	
Includes new BIOS and UTIL files.			
SP0100	400 Series Development Update to p-System IV.20.....	\$ 100	
Includes new BIOS and UTIL files.			
*** Requires prior Development purchase.			
SP0100	400 Series BIOS Source (B diskette).....	\$ 80	
Source to the original user PROM, BIOS, UTIL and boot programs.			
880140	Sealed/Used 87-80 Timberline Spreadsheet SP80.....	\$ 35	
*** Requires prior Spreadsheet purchase.			
880100	Graphics Update diskette 81 (800000).....	\$ 85	
880140	Graphics Update diskette 80 (80000).....	\$ 85	
*** Requires prior Graphics purchase.			
Call Pacer)	880000	Page CP/8-888 Update Diskette A.....	\$ 70
	880000	Page CP/8-888 Update Diskette B.....	
	880040	Page CP/8-888 Update Diskette C.....	
	880041	Page CP/8-888 Update Diskette UTIL/STY.....	
*** Requires prior CP/88 purchase / Serial			
Call Pacer)	880074	400 Series CP/8-888 Update Diskette A.....	\$ 70
	880075	400 Series CP/8-888 Update Diskette B.....	
	880076	400 Series CP/8-888 Update Diskette C.....	
	880077	400 Series CP/8-888 Update Diskette UTIL/STY.....	
*** Requires prior CP/88 purchase / Serial			
<b>TOTAL \$</b>		_____	

PAYMENT:  VISA  MasterCard  Check Enclosed

Serial # (Required)	Main Use of Computer (Optional)
Model # (Page 11, 17, 420, 440 or 480)	Memory _____ Hard disk _____ Tape _____ Floppy _____ (K bytes) (M bytes) (Yes/No) (1 or 2 drives)

Notes:  
 (List new Sealed cards used and Vendor)  
 Software \_\_\_\_\_  
 (List programs you use regularly)

## WORD7 File Extension

**WORD7** files end in .JL. This extension distinguishes WORD7 files from other text files on the system. If the file is changed from .JL to a normal .TEXT extension, most other programs can then access the file.

Just naming a file .TEXT does not convert it to a TEXT file, however. You can use the type of a file if you do an extended listing in the file.

WORD7 files created under p-System II.13 will be seen as TEXT files. If they are moved to version IV.23, however, they will appear as DATA files. This may cause no problems as long as you are not trying to move the WORD7 file to another program requiring a true TEXT file. The authors of WORD7 will soon have a fix for this minor problem.

Until then, this program will convert the files. The two utilities used are found on the UTILTY2 disk in the UNITY subdirectory.

### UNIX Manual Cards

The Strike Manual cards have changed to work correctly under UNIX. When you 4 to get 80 on the number end of the card.

### Graphics Update

The Graphics Update is a new release with a few bug fixes and Zetaphotex software.

### CPM Update

The CPM Update is 800 new but offered just in case you missed it last time around.

```
PROGRAM (C:\WORD7)
(COMPILE TO WORD7 FILES BY TEXT FILES OR WORD EXTEND)
COPY (C:\WORD7\WORD7.DSK) WORD7
(COPY (C:\WORD7\WORD7.DSK) 01.00000)
```

```
END (C:\WORD7\WORD7.DSK)
COPY (C:\WORD7\WORD7.DSK) WORD7
COPY (C:\WORD7\WORD7.DSK) 01.00000
COPY (C:\WORD7\WORD7.DSK) 01.00000
COPY (C:\WORD7\WORD7.DSK) 01.00000
COPY (C:\WORD7\WORD7.DSK) 01.00000
COPY (C:\WORD7\WORD7.DSK) 01.00000
```

END

```
WORD7 (C:\WORD7\WORD7.DSK) 01.00000
```

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

WORD7

## Disk Specs

We get asked this a lot, so here are the specifications of the various disk drives used by Oracle. More, now, and in the past.

In the chart below, the drives shown with an asterisk are those currently offered on the 400 Series.

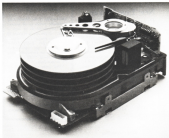
Capacity for both the 18 tracks per track format and for the 14 tracks per track format is given. Remember that UNIX can only operate with the 14 tracks per track format. The p-System and CP/360-800 are capable of operating with either format.

On the 400 Series the system map area required is 1 track x 14 tracks per track. The user area is not shown, but can be calculated with this equation:

$$(\text{Tracks} - 1) \times (\text{Tracks} - 1) \times (\text{Tracks} - 1) = \text{User Area}$$

Large systems use a smaller map area, 1 x 14 tracks per track.

Note that, in general, the bigger the drive the faster it works.



Vendor	Model	Access Avg	Max	Time Ta:Ts	Cyl	Bits	Tracks	Capacity 18 bits/track	14 bits/track	18 bytes/track	14 bytes/track
T&T	300048	88	100	18.0	3008	8	812	8.00	8.00	8.00	8.00
T&T	301048	88	100	18.0	3008	4	1224	12.00	10.00	12.78	10.78
T&T	301448	88	100	18.0	3008	8	1836	17.00	14.00	18.12	14.12
SEIATE	ST410	82	90	18.0	3008	4	1224	12.00	10.00	12.78	10.78
SEIATE	ST410	82	90	18.0	3008	8	1836	17.00	14.00	18.12	14.12
QANTUM	Q160	42	90	12.0	812	4	2048	18.00	14.78	21.00	16.00
QANTUM	Q160	42	90	12.0	812	8	2072	28.00	20.18	31.00	21.00
QANTUM	Q160	42	90	12.0	812	8	4000	38.00	30.00	42.00	32.00
SEIATE	ST410	82	170	20.0	3008	4	1224	12.00	10.00	12.78	10.78
SEIATE	ST410	82	100	20.0	812	4	2048	28.00	20.18	31.00	21.00
MASTOR	XT-1084	80	88	8.0	818	7	8428	62.00	52.00	68.00	58.00
MASTOR	XT-1184	80	88	8.0	818	11	10088	68.00	58.00	74.00	64.00
MASTOR	XT-1140	80	88	8.0	818	10	10070	100.00	82.00	108.00	88.00

\* These are the drives currently offered on the 400 Series.

---

## New Tape Backup Program & Update

Stride has updated the ATMS QIC-80 backup program. A new tape program, which does disk partition backups, is now available as an update for \$11. You will also need to order the BIOS update disk. See order form in this issue. Before installing, contact Stride's Technical Support group, (800) 499-0888, to check the revision level of your board.[]

---

## Work-Around For Timberline SS

### Save Option Bug

While testing software under the new IV.1 p-System release, a bug was found in the Timberline Speedloader. The bug is documented in the speedloader manual but is fairly obscure. It occurs under all versions of the p-System.

The problem occurs when a large spreadsheet has been edited and the user tries to save it. If there is not enough room to save the file, it hangs the system. Typing a Ctrl/0 will return to the operating system, but the edit session has been lost.

Even if you have a big open area before editing, hangups can occur.

Work-arounds: To avoid this problem, save your edit under a new name. Change old copies before starting a session or when you need room.

### Ways WY-80 Spies

For those buying the new Noble terminal or the Ways WY-80 terminal, a new SPIC file is available as an update disk. (See page 61.) Note that speedloader created under one terminal cannot usually be read or updated under a new SPIC file. []

### Yag's California Earth Quake Hot Line

The tragedy of the Mexico City earthquake makes us all wonder what we should do in such a crisis. If you are a resident of California, you can now find out.

Call toll-free 1-800-7-CAL-5353 and get up-to-date information on earthquake safety from our Special 1-800-7-CAL-5353, or TV action films.

YAGS works with a SAGE IV computer and voice records to give you a spin on earthquake safety. If you have a touch-tone phone, you answer yes or no by pushing the 1 or 2 key. YAGS lets you know if you answered correctly.

The station is operated by the California Office of Emergency Services.

---

## Circuit Board Poster Available

Some of you liked the cover of the new In Stride Test Notes so well that you asked where we got the drawing. The circuit shown is the 800 Series Workcenter board, drawn by Stride's CAD/CAM planner.

Due to the interest, Bob Neelham, Stride Founder and CAD/CAM boss, has made this handy offer: Stride will custom draw (on the poster) a modifying/printout, size 18" x 24", of either the CPU or Workcenter board for \$5. The poster has a 1" white border that can be trimmed to needed for mounting and framing. See the order sheet in this issue.

Note the offer is open only as long as Stride has the posters and time to run the printer! Thanks, Bob! []

---

## ASE File Lookup

The ASE editor has become the programming editor of choice among the p-System users. Although the standard p-System editor is a fast reader of ASE and has many of the same commands, ASE has more powerful features. For example, ASE will handle large text files up to 100,000 bytes long (the limit of the file system), search edit and file lookup.

The file lookup is truly useful, you don't have to leave the editor and go to the files when you're looking for the names. ASE will bring up a list of the files in the volume and let you select the one you want. However, differences exist between ASE 1.8 for the IV.1 p-System release and ASE 1.8 for version IV.28 and IV.21. The file lookup facility of ASE 1.8 under IV.11 does not always use files created by ASE 1.8 and the standard p-System editor (SDE) under IV.21.

If you receive some ASE 1.8 files but are using ASE 1.8 you can make the files visible again. Use the instructions in the file: C:\change.wy. This changes all of the files as needed but their original names are kept.[]

---

## People & Products

David G. Bark currently runs the symbolic manipulation program REDUCE 3.8. He's interested in talking to others working with REDUCE and perhaps starting a users' group. If you're interested, you can reach David at: Department of Physics, 940 White Street, Worcester, MA 01090 (617) 940-7100.

DEEMaster is a database program distributed by Interware. An Interware version of the p-System-based directly, a different version of DEEMaster is needed for each p-System version: IV.11, IV.28 and IV.21. Versions are currently available for IV.11 and IV.28 but not for IV.11 at this time. A new version is in the works by author Jerry Mason and we will let you know when it is available.

BULK RATE  
U.S. POSTAGE  
PAID  
PERMIT #888  
RINDI, NEWYORK

**Editor:** *Verlan Jovanovic*

*In Stride Test Notes* is a publication of Stride Micro, issued eight times yearly. Subscriptions are \$12 for one year and include the *In Stride* paper magazine which is published quarterly, for a total of 12 issues per year.

Each *Notes* back issue is also available for \$6.00 and *In Stride* back issues for \$1.00 on supply limits.

Purchase of a Stride computer includes a one-year subscription upon receipt by Stride Micro, Inc., of a fully completed owner's registration card.

Requests for subscriptions, reprint permissions, ad rates, bulk orders or calculation of prospective articles should be sent to the *In Stride* Editor at Stride Micro's New address.

Postmaster: Change of address notices should be sent to:

*In Stride* Editor  
Stride Micro  
P.O. Box 20028  
Reno, NV 89529-0028  
(702) 822-8888 (Mon.-4 p.m. PST)  
TWX: 610-393-8021

*Stride Micro Eastern Division*  
111-116 Washington Street  
Sherborn, MA 01940  
(617) 829-0750

*Stride Micro Southern Division*  
13560 Nord Road  
Suite 200  
Dallas, TX 75249  
(214) 242-7828

*Stride™*, *Performance By Design™* and *MSD™* are trademarks of Stride Micro.

©Copyright 1981-1983 Stride Micro  
All Rights Reserved Worldwide

**STRIDE**  
MICRO