CROSSTALK
Journal of Hewlett-Packard Technical Computer User Groups **June 1982**

COMMENTS FROM H.P.

Why did we decide to call this newsletter "Crosstalk"? It is, afterall, a newsletter for members of our technical computer user groups — so why Crosstalk?

At the moment, there are three active technical computer users groups in Australia. A HP1000 Users Group in Sydney and a HP1000 Users Group and Desktop Computer Users Group in Melbourne. The main aim of these users groups is to provide a means and the environment to enable users to help each other by exchanging ideas, hints and experiences amongst themselves. This is usually done at regularly scheduled meetings. However, there are a significant number of users who would like to participate, but because they do not reside in Sydney or Melbourne, they cannot easily attend these meetings. Also, the three users groups operate independently of each other.

It became evident that what we needed was a vehicle to carry information between users groups and also to those users who wish to belong to a group, but cannot attend their meetings. Hence "Crosstalk" was born. Crosstalk will contain information of interest to all users of Hewlett-Packard technical computers. The information will consist of articles contributed by the users themselves, by members of Hewlett-Packard and, where relevant, by people from the industry. During its development phase, Hewlett-Packard will act as coordinator, gathering the inputs from the users, arranging the printing and returning copies to the users groups for distribution. We anticipate that once it is underway, the users may want to take over the whole process.
The success of this newsletter is in the hands of the users. Unless the users contribute, there will be little to print. Unless users comment, then it will not improve, and unless the users are enthusiastic, it will fade away.
We believe this newsletter will quickly become a valuable and sought after publication and we are pleased to have been part of its development.

Dennis Malseed
Australasian S.E. Manager (Acting)

COORDINATOR'S COMMENTS

Greetings. Just a few words to explain my role in the production of Crosstalk and my expectations of you, the readers and contributors.
Rather than each of the Hewlett-Packard technical computer user groups producing their own newsletters, it has been decided to combine the newsletter contributions into this single publication. Although some of the articles from a group may hold little interest to users outside that group, this should not deter you from contributing and should be well outweighed by the benefits of having more information to share with more users.
It is not my role to determine what should and should not be published. This and other editing tasks will be performed by your own group editor, to whom you should send any information to be published. The editor of each group will collect that group's contributions and forward them to me for each edition. My task is therefore to coordinate and combine the contributions from each of the groups and pass them on to the printer in a suitable format.

I am new at this task, so I would greatly appreciate any comments, suggestions or constructive criticisms from you. I would like to have Crosstalk evolve until it is of a form which is of most use and interest to the most readers. To achieve this I must have feedback from you. This is your newsletter so get involved and make it of value to you.

Glenda Patterson
Crosstalk Coordinator,
Hewlett-Packard, Melbourne.

CONTENTS

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

Now you can get high speed, dual density tape drives for your HP-1000L Series

Dylon Series Ten Gives You:

Dual Density (800/1600 B.P.I.)
7", 8½" or 10½" Reels
Standard or Vacuum Column Versions
Rack mounting or Packaged Styles
Transport speeds up to 125 ips

And

They all plug right into your HP-IB interface with complete RTE/L and RTE/XL operating system compatibility.

DATAMATIC PTY LTD

SYDNEY 438 2699 MELBOURNE 818 0674
ADELAIDE 297 8510 BRISBANE 31 2350
PERTH 382 1744

This is only one of many Dylon products available. Consult Datamatic for your HP-IB Tape requirements.

DATA BASE INTEGRITY

. the Australian Consumers' Association we maintain several Image 1000 data bases, one of which occupies 120 megabytes. The data bases are used for a subscription service for our magazine CHOICE. As the data is used for mailing information it is important that it be accurate for every subscriber and that all the nuts and bolts are in the right place. To assist in this several utilities were written to augment the IMAGE 1000 system to enhance checks on data storage integrity. The following are some thoughts on the subject used in designing such utilities.

This system consists of a suite of programs and subroutines which permit specialized interrogation of the Data Base. The programs are aimed at checking the physical integrity of the data and control information maintained by the Image 1000 system. Checks are performed on record counts, key item values, and the linkage structures. In the past it has been found that data can become destroyed or become inconsistent through problems such as powerfails during data base updating and from other as yet unexplained phenomena. With our data base, like many others, the effect of even a single bit error in ' ' data can be a serious problem. To .intain the integrity and usefulness of our data resource it is necessary to be able to quickly detect and repair any errors. With Image, as the physical binding of the data is very tight using many different linkage structures to bind the data, the detection of all possible errors can be difficult.

With the utilities which are supplied with Image there is not much in the way of error detection facilities and only a very roundabout solution to repair. The utility DBSPA is useful but it is very limited in the type of errors it can detect and it doesn't supply any ancillary information should it detect an error. Briefly DBSPA confirms that the used record counts maintained in the root file are correct by sequentially reading all the data sets. For each set it counts the used records and then compares the count with the root file. The fact that DBSPA may find no difference between the number of used records and the root file counts doesn't necessarily mean the data base is without error. A common error such as a broken detail record chain would go unnoticed.

For the repair of damage to the data base using the utilities supplied with IMAGE 1000, the only solution is to perform a data base unload, reload. A rebuild cannot of course correct data which has become corrupted but it will regenerate all the linkage structures of the data base. However this solution can be very time consuming if the data base is of significant size (► 10 megabytes) and has a reasonably complex logical structure. Given there may only be one error in the data base this is not a very appealing solution.

A better solution for repairing damage would be a program allowing you to edit data base records, word by word. Leaving this program to your imagination the problems still remain though of adequate error detection and error reporting so that repairs can be made. A more general program(s) than DBSPA is required. The program or programs would hopefully not only confirm the record counts but also check the linkage structures as comprehensively as possible and on detection of an error dump as much useful information as possible to the user. In trying to achieve these goals

three programs have been written, SCHAN, SFRC and SDM. Briefly SCHAN is used to check the linkage structures of the detail records, SFRC checks the free record chain and SDM checks the linkage structures of the master records.
To test the linkage structures, aspects of sequential and direct record access are used to isolate any potential problems. The basic idea is to sequentially read a file and for each used record that is encountered, to try and reach the same record via direct access. Consider a master data set, each record has a key value which is used to calculate the records relative address in the file (this forms the basis of hashing). A simple test to apply to such a set is to sequentially read the set and for each used record take the key item value and calculate the expected record address using the hashing algorithm. This calculated address should of course match the current sequential read record address. Checking a master set in this way from beginning to end should confirm that all the synonym chains are intact and that the key item values have not been corrupted. This technique forms the basis of SDM. The program SDM can also check that for every detail record there is a master record with the correct key item value. To perform this test the detail set is sequentially read and the relevant key item value is extracted and used to calculate the position of the master record in the master set.
Program SCHAN uses this sequential/direct access technique in a slightly different way. Schan is used to confirm detail record chaining. A sequential read is performed on a master set for which the detail record chain is to be checked. For each used master record, chain information maintained in the media section of the master record is used to determine the chain head, chain foot and length of the detail chain to be checked. Starting from the chain head a chained read is performed on the detail records. As the chain is read the forward and backward pointers are checked and the value of the key item for this chain is checked against the key item in the master record. When the whole chain has been read the number of detail records on this chain is checked against the chain length as kept by the master record. This program requires some specialized routines to extract the correct chain information from the master record and perform a special chain read on the detail records. Program SFRC tests the free record chain. For a given data set the root fi' is accessed to obtain the address of th.. next free record and the expected number of free records. SFRC then performs a chained read of the free record chain confirming its linkage structure and the number of free records.

Using the programs

Probably the most comprehensive program is SCHAN. Not only does it check the detail chains of a set but it also checks record counts of the relevant master and detail sets. In the case of a missing master record, SCHAN will at least give a warning of the problem. Also SCHAN offers a trace mode for checking individual detail chains. If a key item value is specified in the run command, a directed read is performed on the master set to obtain the relevant master and hence the required detail chain information. In this mode a dump of record pointers and detail record contents is output to the list device. In the case where it is believed a master record is missing then of course SDM would be used. How often the programs should be used is of course dependent on the particular application of data base. It is suggeste though, that should there be say a power fail during operation of the data base then at least SCHAN should be used to check a few of the chains. Now armed with your test and edit programs you should have at least some chance against the occasional glitch.

Australian Consumers' Association
Chris Doyle
May '82

WOULDN'T IT BE NICE ...

- If FMGR ST & DU didn't have the nasty enhancement of truncating your records to 128 words without telling you about it.
- If the LI (list) didn't truncate screen listings to 80 characters.
- If WILDCARD NAMRS were allowed in most FMGR commands (especially PU).
- If all utility programs had a similar way of exiting or aborting such programs (e.g. A, 1A, E, 1E, EX, EN, END, etc.).
- To be able to do a new system generation in a few minutes instead of the 6 plus hours it takes now (not counting the hours required to LOAD and SP all your own private programs).
- If LU numbers could be made to point to devices on the HPIB with secondary addressing capability (e.g. HP 2240, HP6940B, Zeta Plotters, etc.)
- If the PASCAL library contained some decent string handling routines a la HP 9826 PASCAL.
- If the shortest PASCAL program didn't take 10 pages (don't tell me about compiler options).
- If there was RTE6 VM for code.
- If the programmers reference manual was updated to show the calling parameters for all those system routines that you know exist but can't get your hands on.
- If all the manuals were provided on a mag tape and all their indexes were in a disc file which would be searched with a program supplied, e.g. where do I find a routine to load segments.
  : RU, WHERE, SEGMENT

QUESTIONS AND ANSWERS

How? Why? Where? Who?

Have you ever considered adapting your computer system to some new application and wished you could first get comments and opinions from someone who has tried something similar to avoid possibly wasting time and effort?
Do you have a problem which you think some other user could help you with, if only you could contact that user?
If you are considering new hardware, would you be interested in the comments of someone who has such hardware currently in operation?
For the above and many other reasons, you may find help in this section of CROSSTALK in future. Explain your question, problem or interest here and the wait for the next edition in eager anticipation of a response from a fellow reader. If you can suggest a solution to any questions appearing in this section, take the time to help a fellow user by writing a response for the next edition. You don't have to be an experienced journalist; just a few simple lines would be greatly appreciated. This is your newsletter, make sure there is something in it for you!

WHAT'S IN A NAME?

After considering the hilarious possibilities implied by deleting the word 'Computer' from our name, "THE H.P. DESKTOP COMPUTER USERS GROUP OF VICTORIA" was officially launched with the adoption of our constitution on Tuesday, May 25th at H.P. in Blackburn.

At this our second general meeting we worked right through a drfaft constitution pushing it into a final form which should serve us well in the forthcoming ?rs. Copies will be available by the ?xt general meeting.
A subscription of $10 p.a. was decided on at the meeting and under the terms of the constitution it will fall due on 1st March each year.
A number of subscriptions were paid on the spot!

H.P.D.C.U.G.V. MEMBERS

Following the adoption of the Constitution, your annual subscription fee of $10 is now due and payable.
Send your cheques to -
Ron Davis,
Rossmore House,
Ridge Road,
MT. DANDENONG, 3767

THIS IS YOUR GROUP!

Glenda Patterson of H.P. gave a short presentation about her work on "ROSSTALK" to date and this led to a ?ll for the position of Newsletter Editor for the group. It was then proposed that the job be split to reduce the workload and inaugural positions went to:
Rod Johnston - Advertising Editor
Bernie O'Shannessy - Technical Editor
By this time the temptations to try the refreshments provided by H.P. proved too overpowering and we paused for a snack before settling down to listen to a talk by Alan Stoops of H.P. With the aid of a series of well prepared slides Alan introduced us to the concepts of using the G.P. I.B. with H.P. instrumentation and computers to handle problems of testing measurement and control. Judging by the number of questions asked, Alan can count his talk most successful.

The meeting wound up with a further visit to the refreshment tables and general informal discussion.

Bernie O'Shannessy

The HP85's revolutionary CPU

Occasionally I receive telephone calls from people, usually interested in purchasing an HP85, who ask me "what sort of CPU has it got?" The short answer I give is a "Hewlett-Packard special", and then try to briefly describe its attributes. The article following was taken from 'Communicator' magazine, July '81,

Cont. next page

H.P. DESKTOP COMPUTER USER'S GROUP COMMITTEE

Here is a list of the newly-elected committee members for the Victorian desktop user's group and a brief description of their role. Please do not hesitate to contact the appropriate person for any enquiries or to pass on comments.

President/Secretary:	Bernie O'Shannessy, 840 1222 General presidential duties, chairman of meetings, general correspondence.
Membership Secretary:	Ian McWilliam, 819 8864 All enquiries regarding mailing list for group, new memberships/application form, etc.
Treasurer:	Ron Davis, 751 1526 Payment of membership subscriptions (now due!), all finance.
Librarian:	Tony Stevens, 630 7981 Tony is custodian of all manuals, magazines, catalogues, etc., and keeps the membership details. So if you want to know who else has a 9845 together with a 'Forecasting Pac' or similar information, give Tony a ring.
Newsletter Editor:	Bernie O'Shannessy, 840 1222 Our poor hard-working President/Secretary is also in charge of newsletter editorship! Please direct all those articles, questions and comments, etc. for the newsletter to Bernie. Remember, the newsletter will be as good as you make it!
Advertising Editor:	Rod Johnson, Please submit all display advertisements (ready for publication) to Rod for publication in the newsletter. Also any classifieds, wanted to sell, buy, exchange, etc. (free of charge to members). Mail to: ADVERTISING EDITOR, H.P. DESKTOP COMPUTER USER GROUP, C/- 47 BURSANIA AVENUE, FERNTREE GULLY, VIC. 3156
General Committee Member:	Chris Simpson, 859 6643 Chris has no designated task. If you have any other enquiries, see Chris.

and should explain the situation in more detail.

The architecture is basically different from that of any other CPU. An accumulator, which is used often in other CPUs, is notably absent. It has multibyte internal registers that allow direct manipulation of operands up to 64 bits in length.

Why does HP use its own custom CPU?

Accuracy

One of the most important reasons is accuracy. Where other computers do binary arithmetic, the HP85 does its arithmetic in BCD (bindary coded decimal). Our calculators do BCD arithmetic, and we've spent years testing and perfecting those algorithms, so we KNOW that the answers returned by the HP85 are accurate to 12 digits. Real number calculations in the HP85 are performed internally to 15 significant decimal digits and rounded to 12 digits for presentation.

For all the algebraic functions (+,—,x,/, and SQR), the error is no bigger than one-half count in the 12^{th} significant digit, with correct rounding in all situations. For some rational operations such as RMD and MOD there are no errors whatsoever, regardless of the magnitude of the arguments.

Speed

There are a couple of ways to build a CPU — make it simple and run it fast, or make it complex and run it slow. Most manufacturers try to drive their CPUs as fast as possible. Four megahertz is typical — the speed is limited by the physical characteristics of the device, like capacitance. We have followed the second strategy, performing many operations during each clock cycle. The clock speed is 613 kilohertz!

But, execution speed is more than just a function of hardware design, and the HP85 executes programs on the same order of time or faster than other personal computers on the market.

HP85 BASIC programs are executed by an interpreter, but the code that is interpreted is very different from the BASIC commands as they were originally entered. As the statements are entered into the HP85, they are compiled to a form of RPN, which can be interpreted more efficiently than the BASIC source statements. One of the things that has traditionally made interpreters slow is that they maintain a table of variables which must be searched at run-time for each variable reference. In the HP85 this problem is solved by preallocation of all variable references. During the allocation process, the variable names that occur in the internal RPN form of the program are replaced by the relative addresses of the variables. Then, at run-time, the interpreter has only to read an address and add it to the base address of the program to determine the absolute location of the variable being referenced. For traditional interpreters the time required to access any variable is dependent upon its position within the variable table. In the HP85 all variables are accessed in exactly the same amount of time. All line number references are also replaced by the relative address of the referenced line during the allocation process, eliminating the need to search for referenced lines at run-time. This allocation process also makes possible the convenient REN command, which renumbers not only the line numbers, but all GOTO and GOSUB statements as well.

Architecture

The custom CPU used in the HP85 incorporates many features currently not found in other microprocessors, such as instructions that operate on data one to EIGHT BYTES in length (simplifying multibyte arithmetic and facilitating string manipulations). It also contains features found in other microprocessors that have proven to be desirable.

Figure 1 is a simplified block diagram of the microprocessor. Notice that the architecture is of the classical textbook variety. The three busses constitute two sources and one destination. Data for the ABUS can come from the external bus or the 64-byte on-chip memory. Data for the DBUS comes only from the on-chip memory. The ALU and shifter are in series and capable of full eight-bit parallel binary and decimal (BCD) arithmetic. Control of the entire CPU is handled by a programmable logic array.

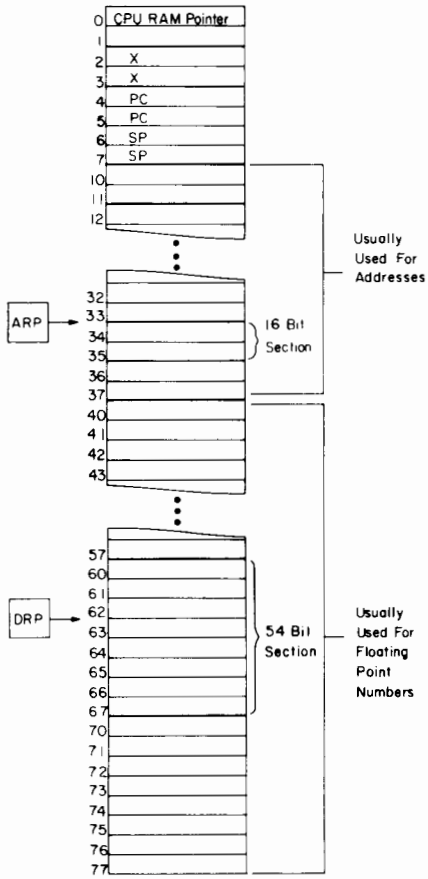


Figure 2

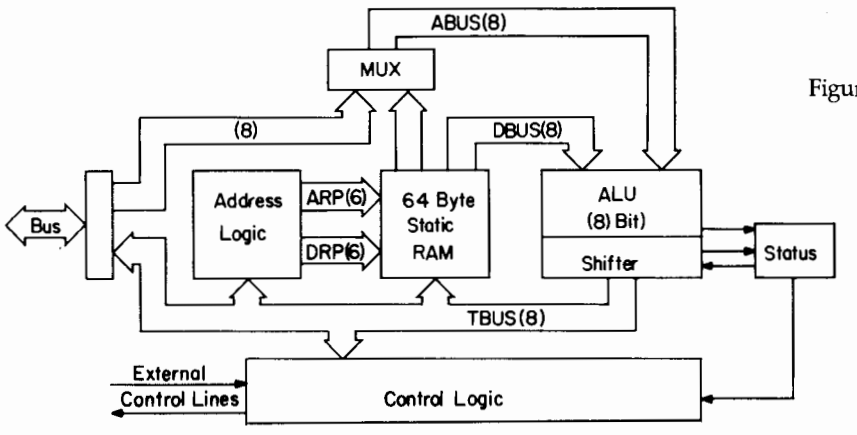


Figure 1

One of the original design goals was to keep the pin count low to minimize package cost, save space, and lower the power requirements (the CPU fits into a 28-pin package). So a single, time-multiplexed, eight-bit bus is used to transport data, commands, and addresses. It turns out to be very efficient for the instruction set implemented.

When more than one byte of data is sent along the bus, an address is needed only for the first byte. The remaining data is assumed to be in consecutive locations. The memory controller and I/O units handle this by incrementing their own address registers upon seeing either a read or a

write instruction. Also, the instruction-fetch sequence was designed so as not to require an updated address when fetching from some consecutive locations. As a result, the bus primarily moves data and instructions and tries to minimize the movement of addresses.

The heart of the CPU is the register bank comprising 64 bytes of RAM. The organization is shown below, with the locations numbered in octal from 0 to 77. The top eight bytes are special purpose registers. Locations 6 and 7 house the stack pointer for the subroutine return address stack. Locations 4 and 5 provide storage for the program counter. Locations 2 and 3 are scratch registers used for index address calculations. Register 0 can be used as a pointer into the registers. The remaining locations are completely general purpose.

The heavy lines in figure 2 are called boundaries. In the first 32 bytes, there is a boundary every two bytes. In the next 32 bytes, there is a boundary every eight bytes. The purpose behind this partitioning is simple: 16 bits are essential for address manipulation, and 64 bits are handy for representing a floating point quantity. The register array is therefore capable of holding up to four floating point numbers and twelve 16-bit addresses. The advantage of 16-bit addresses is, of course, that page boundaries are completely eliminated — the programmer can get at any of the 2^{16} locations with any addressing mode.

The 64-byte register bank is designed as a two-read, one-write memory. Two independent locations from the memory are read at the beginning of a cycle and new information is written into one of those locations at the end of the cycle. THIS ELIMINATES THE NEED FOR AN ACCUMULATOR, WHICH IS A COMMON BOTTLENECK IN MANY MICROPROCESSORS.

Two different address pointers are therefore associated with this memory. The address register pointer (ARP) and the data register pointer (DRP) are independent six-bit locations. Both the ARP and the DRP can be used to address any of the locations in the CPU register bank.

Instructions may be either in a "multi-byte" mode or in a "single" mode. As the name suggests, a "multibyte" operation involves a string of bytes rather than just a single byte. The important point is this: The string may consist of from one to eight consecutive processor locations. The actual locations involved in a multibyte operation are those inclusively between where DRP points and the next boundary. The next boundary is the one in the direction of increasing addresses.

The following examples should help explain this idea.

1. A multibyte increment with DRP set to 70 results in an increment of the 64-bit quantity stored between locations 70 and 77. Higher addresses always refer to more significant data.
2. A multibyte test with DRP set to 44 results in status being set according to the data found in registers R44, R45, R46, and R47. Location R47 is the most significant byte.

If an instruction is not in the multibyte mode, it is necessarily in the single byte mode. In this case the byte referenced by DRP is the only consideration. A single-byte negate with DRP set to 62 negates R62.

So far, only monadic operations have been used as examples. What happens if a two-operand instruction is executed in the multibyte mode? Here is where the CPU really shines! In this case, ARP points to one operand and DRP points to the other. DRP still determines the number of bytes for its operand. The other operand consists of the same number of bytes starting with the location pointed to by ARP. If data is written into a register at the end of a cycle, it goes into the register the DRP Points to. For example:

1. A multibyte add with ARP set to 50 and DRP set to 60 results in the 64-bit quantity starting with R60 being added to the 64-bit quantity starting with R50. The sum is stored in R60 through R67.
2. A multibyte load with ARP set to 11 and DRP set to 74 transfers four bytes, beginning with R11, to locations R74, R75, R76, and R77.

Footnote:

The above article was taken from 'Communicator' magazine and edited by Philip Greetham, Systems Engineer with Hewlett-Packard.

PROGRAMMING TIPS

Cont. next page

Inverse Graphics Display (9845)

A previous programming tip described a procedure for making the initial background of the 9845's display screen inverse. That tip made it necessary to change the pen control from *PEN 1* to *PEN —1* to plot on the inverse background.

If your 9845 is equipped with the I/O ROM together with the Mass Storage ROM, it is possible to change the already plotted graphics in normal procedure into inverse video display.

In line 50, the whole area of the graphics is stored in G(*). Then in line 70, each bit of G(*) is inversed, that is, 1 to 0 and 0 to 1. In line 90, the inversed graphics are loaded back to the display.

```
10  SUB  Inv
20  OPTION BASE 0
30  INTEGER G(16380)
40  G(0)=-1
50  GSTORE G(*)
60  FOR I=1 TO 16380
70  G(I)=BINCMP(G(I))
80  NEXT I
90  GLOAD G(*)
100 SUBEND
```

K. Kanemaru, Cable Research Laboratory
Hitachi Cable, Ltd.
5-1 Hitaka-machi
Hitachi-shi, Ibaraki 319-14
Japan

STOP PRESS .....

The next meeting for the Desktop Computer User's Group will be held on Tuesday, 6th July at 4.30. The meeting will be held at the Shoppingtown Hotel in the Panorama Room. Highlight of the meeting will be a short presentation by Dennis Vetter who is manager of International Marketing & Support of Desktop Computer Division, Colorado. Dennis is here as part of a world-wide seminar, the Melbourne one of which is to be held on Wednesday 7th at the Old Melbourne Hotel. (Most of you will already have received an invitation). Dennis will talk to us about what is happening at DCD at the moment. Following Dennis's presentation HP will buy the first few rounds of drinks, and Dennis will stay a while for an informal chat and drinks with members. All members welcome.

PROGRAMMING TIPS

FOR/NEXT LOOPS IN FORTRAN (HP1000)

Have you ever had the hassle of devising explicit incremental instructions for loops? The looping subroutines shown can be adapted equally well for type 'real' or type 'double precision' (3 or 4 word).
The FOR/NEXT-loops work with positive or negative increments, and like DO-loops, will execute at least once. On completion, the loop variable's value is always 'outside' the loop limits.
A pair of FOR/NEXT subroutines may be used many times throughout the program, but a difficulty arises if they need to be nested. The nesting difficulty is overcome by having pairs of routines called 'FOR1/NEXT1', 'FOR2/NEXT2', 'FOR3/NEXT3', etc., or as many as are required. I have found that three sets suffices even the most amazing programs.
The calls consist of:

```
CALL FORn ( var,from,to,step )
          \    \    \ \     \-increment (+ve or -ve)
           \    \    \ \-termination value
            \    \    \-initial value given to 'var'
             \    \-loop variable
              \-n is part of the name of the subroutine

CALL NEXTn ( var,from,to,step )
```

Line numbers are not necessary!

EXAMPLE 1: A single pair of routines used in different parts of a program.

```
      FROM = 1.
      TO   = 100.
      STEP = 2.
      SUM  = 0.
      CALL FOR0 ( VALUE, FROM, TO, STEP )
          :
          :
      CALL NEXT0 ( VALUE, FROM, TO, STEP )
          :
      B = 25.
      C = -25.
      D = -5.
      CALL FOR0 ( A, B, C, D )
          :
      CALL NEXT0 ( A, B, C, D )
```

EXAMPLE 2: Nested loops, using FOR0/NEXT0 and FOR1/NEXT1.

```
      CALL FOR0 ( A, PI, -PI, -PI/10. )
          :
        FROM = 0.
        TO   = 100.
        STEP = 2.
        CALL FOR1 ( BIT, FROM, TO, STEP )
            :
        CALL NEXT1 ( BIT, FROM, TO, STEP )
          :
      CALL NEXT0 ( A, PI, -PI, -PI/10. )
```

Here is a brief schematic of how they work. Notice that the CALL to FOR0 is actually executed only once, while a normal exit to NEXT0 again only occurs once. The jumps to re-perform the loop occur "out-of-view".

```
:
:
(program code)                ("SUBROUTINE FOR0"
:                             (     ASSIGN 10 TO LABEL
CALL FOR0  -------------->> (
    (return address)  <<--- (  10 RETURN  <<--------------
    :                                                     |
    :                                                     |
    (loop code)                                           |
    :                         ("SUBROUTINE NEXT0"         |
    :                         (     INCREMENT var         |
CALL NEXT0  ------------->> (     IF END-OF-LOOP? GOTO --
(return address)  <<------- (     ELSE DO A NORMAL RETURN
:
```

These routines were originally developed as part of a subdivision plotting routine, where we needed to draw segments of arcs incrementally at any orientation and in both clockwise and anticlockwise directions. They reduced the amount of IF'ing and ELSE'ing to nill.

Here is a copy of the code:

```
FTN4
C-------------------------------------------------------
      SUBROUTINE FOR1( A, B, C, D),16-10-81 S: For...
      COMMON /FRNX1/ LABEL, E
      A = B
      E = C*D
      ASSIGN 10 TO LABEL
10    RETURN
      END
C-------------------------------------------------------
      SUBROUTINE NEXT1( A, B, C, D ),16-10-81 S: ...Next
      COMMON /FRNX1/ LABEL, E

      A = A + D
      IF ( A*D .LE. E ) GOTO LABEL
      RETURN
      END
C-------------------------------------------------------
      SUBROUTINE FOR0( A, B, C, D),16-10-81 S: For...
      COMMON /FRNX0/ LABEL, E
      A = B
      E = C*D
      ASSIGN 10 TO LABEL
10    RETURN
      END
C-------------------------------------------------------
      SUBROUTINE NEXT0( A, B, C, D ),16-10-81 S: ...Next
      COMMON /FRNX0/ LABEL, E
      A = A + D
      IF ( A*D .LE. E ) GOTO LABEL
      RETURN
      END
C-------------------------------------------------------
      BLOCK DATA FN1,16-10-81 B: Return address
      COMMON /FRNX1/ LAB1, E1
      COMMON /FRNX0/ LAB0, E0
      END
```

(The NEXT calls could be simplified by leaving out all the parameters and passing them through COMMON. However, as it stands, it is still a good programming tool.)

Alf Lacis
C/- Chatads P/L
P.O. Box 846
TRARALGON, Vic. 3844

HP1000 Users Group News (Vic.)

The second Annual General Meeting of the HP1000 Users Group was held on Tuesday, 27th April, 1982 at Hewlett-Packard, Blackburn. Those elected as office bearers for the next 12 months were:

John Gwyther	TUSC Computer Systems	(03) 859 9487	President
Norm Kay	CSIRO	(03) 544 0633	Treasurer
Owen Marsh	Dept. of Aviation	(03) 615 8669	Secretary
Glenn Lawrence	Vic. College of Pharmacy	(03) 387 4705	Committee
Chris Emery	Dept. of Aviation	(03) 667 2328	Committee

The guest speaker at the meeting was Ed Brumit, Marketing Manager, Data Systems Division, HP, USA. A very interesting talk was given on the future direction of the HP1000 series computers, including the introduction of the A600 and A700 computers and the planned release of software packages and operating system enhancements. It is hoped to organise a number of one-day workshops later in the year.

These will cover such topics as Languages (Pascal, Fortran 77, Macro), Graphics/1000, Image Query data-base management system, Plus/1000 library software packages, Third party HP1000 software packages, etc.

Subscriptions of $12 for the year March 1982 to February 1983 are now due. If you have not already done so, send your cheque to The Treasurer, HP1000 Users Group, P.O. Box 132, Mt. Waverley, 3149.

Contributions relating to any aspect of HP1000 computer systems (hardware, applications, software, programming hints, etc.) are required for publication in this newsletter. Contributions should be sent to the Users Group at the above address and they will then be onforwarded to HP for publication.

SYDNEY AND THE BUSH-EXPERIENCES WITH THE HP1000 XL SERIES

(A summary of a talk presented at the May meeting of the HP1000 Users Group, N.S.W.)

PROBLEM: To supply some computing power to a fledgling Commonwealth Govt. Statutory Authority split in two. One half in high-rise Sydney, the other in the uranium mining province in N.T., adjacent to Arnhem Land. Applications to vary from environmental database creation to laboratory instrumentation data capture and control.

SOLUTION (mine!): Purchase of two HP1000 XL machines with appropriate peripherals.

PROGRESS REPORT: The XL machines are essentially 16 bit microcomputers with an emphasis on I/O and real-time processing. They are, however, capable of running most of the HP big brother M, E & F series software-IMAGE DBMS, language processors (FORTRAN 4X, BASIC, Pascal), D/S-1000 and GRAPHICS-1000. The operating system (OS) is a subset of RTE-IVB. It has no session monitor, hence, to set up terminals for programme development, etc., one merely RPs FMGR and soforth to that terminal as the primary scheduled programme. One consequence of the simplified OS is the time taken for Sysgen's-typically 5 minutes! There are limitations on programme development in that memory partitions can be no larger than 32 Kwords, hence, without EMA one has to think carefully regarding judicious use of segmentation — no doubt a good thing for modular code but hopeless for handling very large arrays.

The 8 port asynchronous multiplexer is versatile and very controllable programmatically. Its main limitation, for my purposes, is the need to distribute equitably the favours of the two on-board baud rate generators between 8 ports. However in this organisation the connected RS-232 type devices are only used spasmodically thus no major problems have occurred. Devices currently connected to the Mux include an HP2623 graphics terminal, an NBI-3000 word processor, a MICRO-PAD terminal (a device which translates hand-printed characters into ASCII), a 4 pen plotter and an acoustic coupler for remote terminal connection.

Currently on order for the Sydney system is the new 7908 Winchester disc with an integral streaming tape unit. Recently we acquired a Kennedy mag. tape drive with a Dylon controller connected to the XL via HP-IB, and which uses the standard HP driver (to my utter amazement the unit and software worked first time up!)

As yet I have no figures describing degradation of system response with terminal additions, etc., but I do have the subjective feeling that the time-sharing facility could probably handle a number of concurrently-executing programmes (particularly of an I/O nature), provided there was little programme development occurring simultaneously. The XL is, after all, a microcomputer with no floating-point hardware, array processing etc. — although the HP announcement of the A600 and A700 processors point to an obvious growth path.

Generally the machines have been successful (would my ego admit otherwise!). There are a few provisos — the documentation could stand a lot of editing and unifying — there's too much of a hastily-prepared air about them, compared to the usual HP quality regarding hardware manuals. System revisions documentation requires most detail also, but generally, the XLs have performed creditably and have been a lot of fun.

Bill Wallace
Office of the Supervising Scientist for the Alligator Rivers Region

Plus/1000 Library

I receive calls on a regular basis from people requesting details (and programs) from the Plus/1000 library. Some of these callers insisted that I immediately send them a complete copy of the library and all supporting documentation.

What is the Plus/1000 Library?

There used to be a library of contributed software called LOCUS (which probably stood for Library Of Contributed User's Software).

This library was maintained (in a manner of speaking) by Hewlett-Packard and was sold to customers for about $500. There came a time when the library was in such a sorry state

that HP decided to re-organise, re-structure and re-name this collection of software.
At about the same time, the HP1000 International User's Group was formed and HP decided to donate the entire library to this new Group (actually it saved them the cost, time and hassles of maintaining this rather large collection of software).

Where can I get the Plus/1000 Library?

For an annual cost of $US250 you can become a member of the HP1000 INTERNATIONAL USER'S GROUP. This entitled you to a mag-tape copy of the library plus a catalog index and abstract of each package in the library. The address is:
Attention Judy Gelden
Membership Records Co-ordinator,
HP1000 International User's Group
289 South San Antonio Road,
LOS ALTOS, CALIFORNIA 94022,
U.S.A.

Membership of the HP1000 IUG also gives you 6 issues of their newsletter per year and library updates twice a year.

Where else can I get the Plus/1000 Library?

The policy of all HP1000 User's Groups is that each member may copy only one contribution. Please note that one contribution can sometimes mean 10 to 15 separate programs. Contact a committee member of your local User's Group to arrange for a copy.
All media and postage costs to be borne by the member.

How do I submit a contribution to the Library?

There is a program called SUBMIT which leads you through the submission process step by step and we will gladly give you a copy.
As a bonus item, any member of the HP1000 User's Group (NSW) who submits a worthwhile contribution to the library, may ask for any two PLUS/1000 contributions.

Corrado Diqual
President,
HP1000 Users Group (NSW)

How do I find out what's in the Library?

If you would like a copy of a condensed index of the PLUS/1000 Library, write to Corrado Di Qual, HP1000 Users Group (NSW), GPO Box 3060, Sydney, N.S.W. 2001.

THIRD-PARTY SOFTWARE FOR HP 1000 COMPUTERS

by George Low, DSD

Over 40 third-party software programs are available for HP 1000 computers via the HP PLUS Program. HP PLUS, an acronym for Hewlett-Packard Program for Locating User Software, provides HP 1000 users access to a wide range of application programs, tools, utilities and languages to complement HP software for successful solutions in factory and plant automation, computer-aided engineering/design, communications, manufacturing management and other applications. Representative programs available for the HP 1000 computers from the HP PLUS third-party software suppliers are listed below.

PROGRAM	DESCRIPTION	COMPANY
CEADS-CADD	Computer-Aided Drafting	Holguin & Associates
NISA	Finite Element Analysis	Engr. Mech. Research Corp.
DISPLAY/DIGIT	Finite Element Graphics	Engr. Mech. Research Corp.
QADM	Gage/Instr. Calibration	Hansford Data Systems
ATLAS/1000E	ATE Test Language	Lexico Enterprises
ATLAS/1000D	ATE Test Language-Digital	Lexico Enterprises
ATLAS-80	Table-Driven ATLAS	Lexico Enterprises
ATLASCAN	ATLAS Syntax Checker	Lexico Enterprises
APT	Numerical Control	University Computing Co.
SDM	Structural Dynamics	Structural Meas. Systems
ILS	Signal Analyses	Signal Technology, Inc.
Many	Microproc. Cross-Assemb.	Microtec
Many	Micropro. Meta-Assemblers	Microtec
Many	Microproc. Simulators	Microtec
SCONS/1000	Source Control System	Corporate Computer Sys.
DELTA/1000	File Difference Locator	Corporate Computer Sys.
FBUG/1000	FORTRAN Debugger	Corporate Computer Sys.
TFORM/1000	Text Formatter	Corporate Computer Sys.
VEDIT/1000	Text Editor	Corporate Computer Sys.
SPELL/1000	Automatic Proofreader	Corporate Computer Sys.
HP/C	C Language Compiler	Corporate Computer Sys.
COBOL/1000	COBOL Compiler	Corporate Computer Sys.
CEMS/1000	Central Energy Mgmt. Sys.	Int'l. Energy Cons. Sys.
GES/1000	Cut and Fill Volumes	Associates II
MATH/1000	Business/Scientific Model.	COMPROG
Terminal Mgr.	CRT Development Tool	Industrial Computer Corp.
Image Inter.	Data Base Handler	Industrial Computer Corp.
PFAS	Pascal File Access Sys.	C.J. Wigglesworth
PDS	Pascal Development Sys.	Theta Business Systems
G/L	General Ledger	Theta Business Systems
A/P	Accounts Payable	Theta Business Systems
A/R	Accounts Receiving	Theta Business Systems
Payroll	Automated Payroll System	Theta Business Systems
O/E	Order Entry	Theta Business Systems
JPICS/CGS	Process Control	Scientific Sys. Services
Prod. Mon.	Manufacturing Simulation	ITP Boston
QADM	Process Analysis	Hansford Data Systems
QADM	Incoming Inspection	Hansford Data Systems

More third-party software supplier programs are being qualified and added continually to the HP PLUS Program. For more information on the above programs and other software programs available on HP 1000 and other HP computers, contact your local HP Sales Representative.

Printed by permission of INTERFACE 1000

Focus 1000

"CMGR/1000 CARTRIDGE MANAGEMENT SYSTEM FOR THE HP 1000"

Submitted by:
C. Boozer, Universal Computing,
P.O. Box 877, Del Mar,
California 92014 U.S.A.

INTRODUCTION

CARTRIDGE MANAGER 1000 (CMGR/1000) is a complete software package for controlling disc space on an HP 1000 computer system. The HP 7905, 7906, 7920, 7925, 7906H, 7920H, 7925H, 9895 Flexible, and 7935/7933" Winchester disc drives are completely supported. The package will operate using either the 13037B/C Multiple Access Controller (MAC) or the 12821A Integrated Controller Disc (ICD) interface. The CMGR/1000 package is fully compatible with RTE-IVB (HP product 92068A) or RTE-6/VM (HP product 92084A).

As the primary disc control within the operating environment, CMGR/1000 provides the user with complete services for cartridge-related functions.

The major features of CMGR/1000 are summarized below:

- — Unlimited Numbers of cartridges can reside on disc
- — Optimized use of disc logical units
- — Ability to change any cartridge directories
- — Tree structured cartridge directories
- — Automatic cartridge archival and retrieval
- — Database management of on-line and off-line cartridges
- — Status reports of disc usage and availability
- — Mag tape and pack compatibility with other computers
- — Installation does not require System Generation
- — Compatible with RTE, FMGR, and the FMP package

METHOD OF SOLUTION

The current File Management Package (FMP) and File Manager (FMGR) formats and concepts are not modified. Cartridges are kept within files, and can be made available to the system via a central program. These files are called "cartridge files", and appear as standard type in files to the FMGR and the system. The database is kept in both a central file, and in distributed nodes contained within each cartridge file.

NOTE:
This is an excerpt from the article appearing in the FEB/MAR '82 edition of 'INTERFACE 1000', the HP1000 International Users Group magazine. For further information, refer to the article in INTERFACE 1000, or contact the International Users Group or C. Boozer.

COMING EVENTS

June 21 RTE-6/VM Upgrade Course. For 2 days at HP Melbourne.
June 21 HP1000 System Manager course. For 5 days at HP Sydney.
June 28 9826/36 Basic Operating and Programming course. For 5 days at HP Melbourne.
June 28 IMAGE/1000 course. For 5 days at HP Sydney.
July 5 RTE-6/VM Session Monitor course. For 10 days at HP Melbourne.
July 6 Desktop Users Group meeting, 4.30 at Hewlett-Packard. Guest speaker Dennis Vetter from DCD Colorado. (To be confirmed.)
July 15 Victorian HP1000 Users Group meeting to be held at the CSIRO Division of Chemical Physics, Bayview Avenue, Clayton, Vic. commencing at 4pm. Norm Kay will discuss the direct linkage of an HP1000 computer to the CSIRONET network. Members and friends are welcome to attend.
July 19 HP Basic Programming course. For 5 days at HP Melbourne.
July 19 RTE-6/VM System Manager course. For 5 days at HP Melbourne.
July 19 RTE-A.1 System User course, HP Sydney.
July 26 RTE-A.1 System Description course, HP Sydney.
Aug. 9 Introduction to the HP1000 course, HP Sydney.

CLASSIFIED ADVERTISEMENTS

FOR SALE

HP DEMO STOCK
HP85A Desktop Computer. $2,992
9874A Digitizer. $12,904
98439A Ass. Exec. & Dev. Rom. $2,025
98046B RS232c Data Com. $1,051
2146/7910HR HP 1000 L-Series Model 10. $23,116
7225/004/17601a Plotter. $3,038
00085-15001 mass storage rom. $156
00085-15002 plotter/printer rom. $156
98035A Real Time Clock. $540
98032A 16Bit/Interface. $525
All enquiries should be directed to:
Greg Sadler
Hewlett-Packard Aust. Ltd.
31-41 Joseph Street,
BLACKBURN SOUTH, VIC. 3130
Phone: (03) 877 7777

FOR SALE

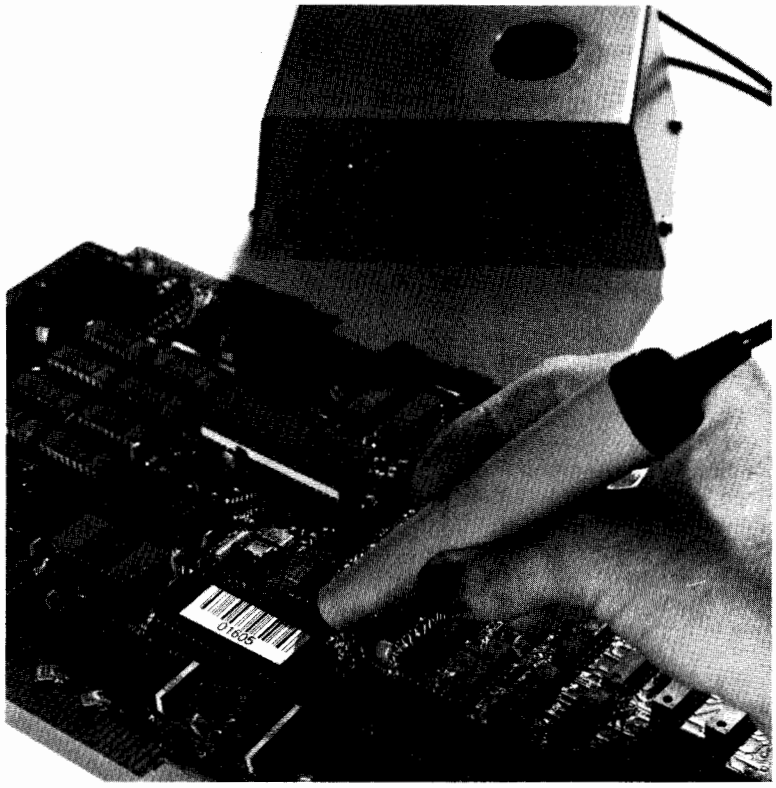
9830 SYSTEM
9830A with 8K memory,
adv. prog. rom, plotter
control rom, ext. I/O
Cassette drive (ext. memory)
to suit 9830.
HP2748b. High speed paper tape reader and interfaces for 9830.
All enquiries should be directed to:
Tab Fried
T.E.D. Engineering Group
Phone: (03) 555 8211

FOR SALE

HP85 including all Roms except advanced programming.
82901M twin 5 ¼" floppy disc drive.
2631B printer.
9111A graphics tablet.
HP125 with BASIC operating system.
Enquiries should be directed to:
Mark Levenspiel
Phone: (059) 85 3731

"I keep six honest serving men,
(They taught me all I knew);
Their names are WHAT and WHY
And WHEN
And HOW and WHERE and WHO."
R. Kipling.

Kipling knew what was needed for Manufacturing Control,
-------Bar Code Data Capture makes it easy for you.



Data capture on the factory floor, in the calibration or QC laboratory, shipping department, or at a point of sale has to be fast, accurate and economical. Databar bar code readers do it all, giving your computer the data it needs and letting you ★

DATABAR

★ *To decode call Harold Norrie or buy a Databar Reader.*

bar code products and systems are designed and manufactured in Australia.

AVAILABLE FROM

NORTRONIC INSTRUMENTS

Box 995, GPO Sydney 2001 Phone: (02) 290-2844 Telex: AA27585