COMPUTER

CENTRE

BULLETIN

EDITORIAL COMMENT

We remind our readers that our programmers have offered to discuss any difficult or unusual problems which might be of general interest, through our new *"Letters to the Editor"* section of the Bulletin. In this way, with your interest and enthusiasm, *"Letters to the Editor"* may become a regular part of the Bulletin. If you have any item of general computing interest, the Bulletin could publish both letter and answer.

As mentioned last month, longer articles on the application of computing to various fields of enterprise and research, would be most welcome. This could include individual work on computing that may be of general interest, or specialised projects carried out by a group of people. If readers and clients of the Computer Centre were to become interested in contributing regularly to the Bulletin, it will surely develop as a medium for the interchange of information and ideas.

The accent of this month's Bulletin is on the PDP 10 Computing System. Two articles are included, one providing helpful information on the use of the computer services, and the other describing FORTRAN memory allocation on the PDP 10. The section on Programming Advice is continued with an advisory flow chart designed to assist the less experienced programmer. A further article on the staff is provided, and new Library Accessions are listed.

STAFF OF THE COMPUTER CENTRE

INTRODUCING TWO LECTURERS

Jim Sokoll holds the position of Lecturer in Computer Electronics within the Department of Computer Science. Jim graduated from the University of Queensland in 1962 with a degree in Electrical Engineering, commencing work that year as a Programmer in the Computer Centre. In 1964, he was appointed to the position of Lecturer.

In this capacity, Jim is responsible for the lectures on Logic Design of Computing Devices, a major subject in the postgraduate Diploma in Automatic Computing. He also instructs second, third and fourth year engineering students in various aspects of computing systems.

Jim is also responsible for the maintenance and development of the GE 225 and the PDP 10 computer systems, and takes care of all matters that affect computer services. In 1968, he attended a course on the construction and maintenance of the PDP 10 at Maynard, Massachusetts.

One of Jim's main interests is to ensure that people, who have a limited knowledge of computers, are able to use the system. He is therefore interested in the modification of both hardware and software to make a computer system usable and simple. His other interests are concerned with compilers and digital communication systems.

John Noad is a Lecturer in Computing within the Department of Computer Science. In 1960, John graduated from the University of Queensland with a Science degree, and, for the following three years, worked as a Demonstrator in the Department of Physics, obtaining Honours in Physics in 1963. He joined IBM (Australia) Ltd. in January 1964, working as a Systems Engineer in both Sydney and Brisbane, where he gained considerable experience in the design and implementation of machine systems, and in the development of programmed applications. In January 1968, he joined the Computer Centre staff.

John's prime responsibility is the postgraduate Diploma in Information Processing, which was offered for the first time in 1968. John lectures in the subjects - Data Processing Equipment, Computer Programming, and Systems Analysis and Design. This course has proved to be so popular that the enrolment for 1969 is three times the enrolment of the preceding year.

INTRODUCING A NEW MEMBER OF STAFF

A new addition to the staff is *John Hendry* who graduated with First Class Honours in Chemical Engineering in 1968.

John is presently employed as a half-time Demonstrator, and is enrolled full-time in the postgraduate Diploma in Automatic Computing. He intends to specialize in the application of digital computers to chemical engineering.

John hopes to go overseas in September to study for his Doctorate in either Canada or the United States. We wish him every success in his career.

RECORD STUDENT ENROLMENT

The increasing demand for computer-oriented courses is apparent from the 1969 enrolments in the Postgraduate Diplomas in Automatic Computing and Information Processing, as there is a very significant increase over the 1968 figures.

In the Diploma in Automatic Computing, there is, at the time of publication, a total of thirty-two students enrolled (five full-time and the remainder, part-time). This compares strikingly with the 1968 total of seventeen. The Diploma in Information Processing is now being offered for the second year, and has an enrolment of thirty students. (Sixteen full-time, fourteen part-time). This is a three-fold increase over the 1968 total of nine.

It is hoped that, with the creation of the Department of Computer Science and the arrival of Professor Rose, an undergraduate computer science degree will be introduced in the near future. This should further cater for the growing demand for courses in this field.

RECENT PUBLICATIONS

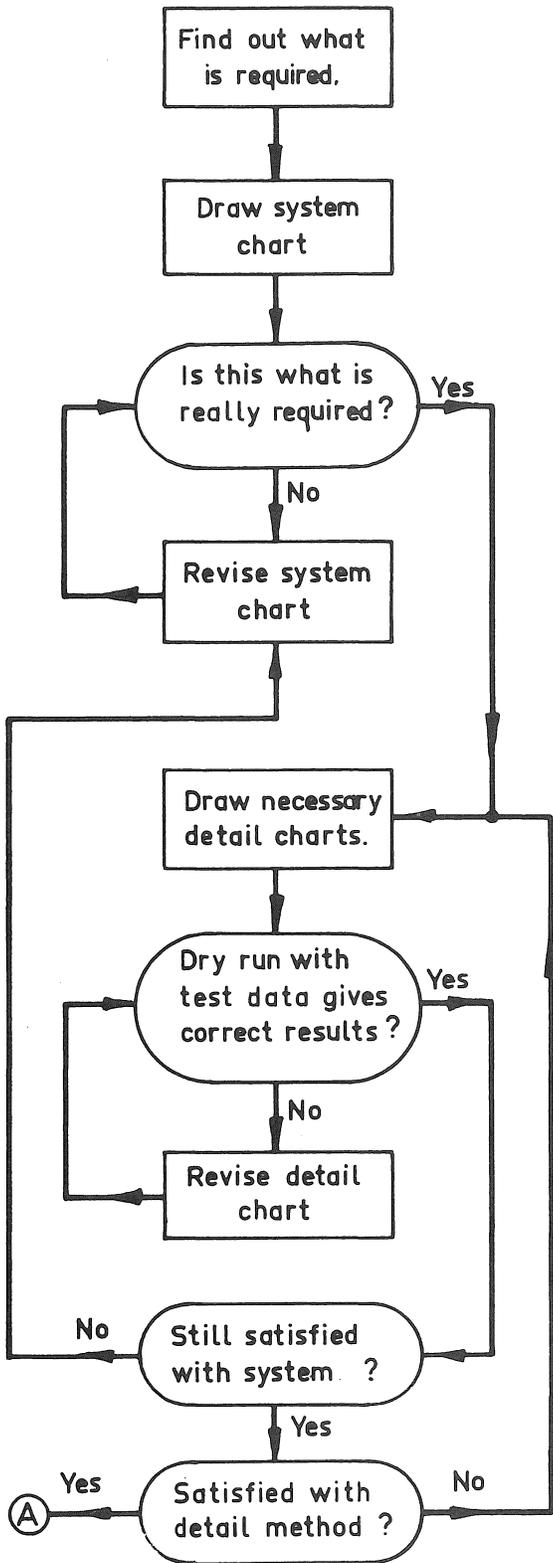The following publication is available at the Computer Centre:

TECHNICAL MEMORANDUM NO. 4: *Character Codes for the PDP 10 Computer System.* R.E. Kelly, E.J. Sokoll.

The establishment of a standard character set and related codes is essential for the interchange of information between computer systems and for representing information within a particular system. The International Standards Organization has recommended a standard code for information interchange which has been substantially adopted for the PDP 10 computer system.

This memorandum defines the character set adopted for the PDP 10, and details those codes used to represent characters on the various recording and storage media.

A GUIDE TO PROGRAMMING
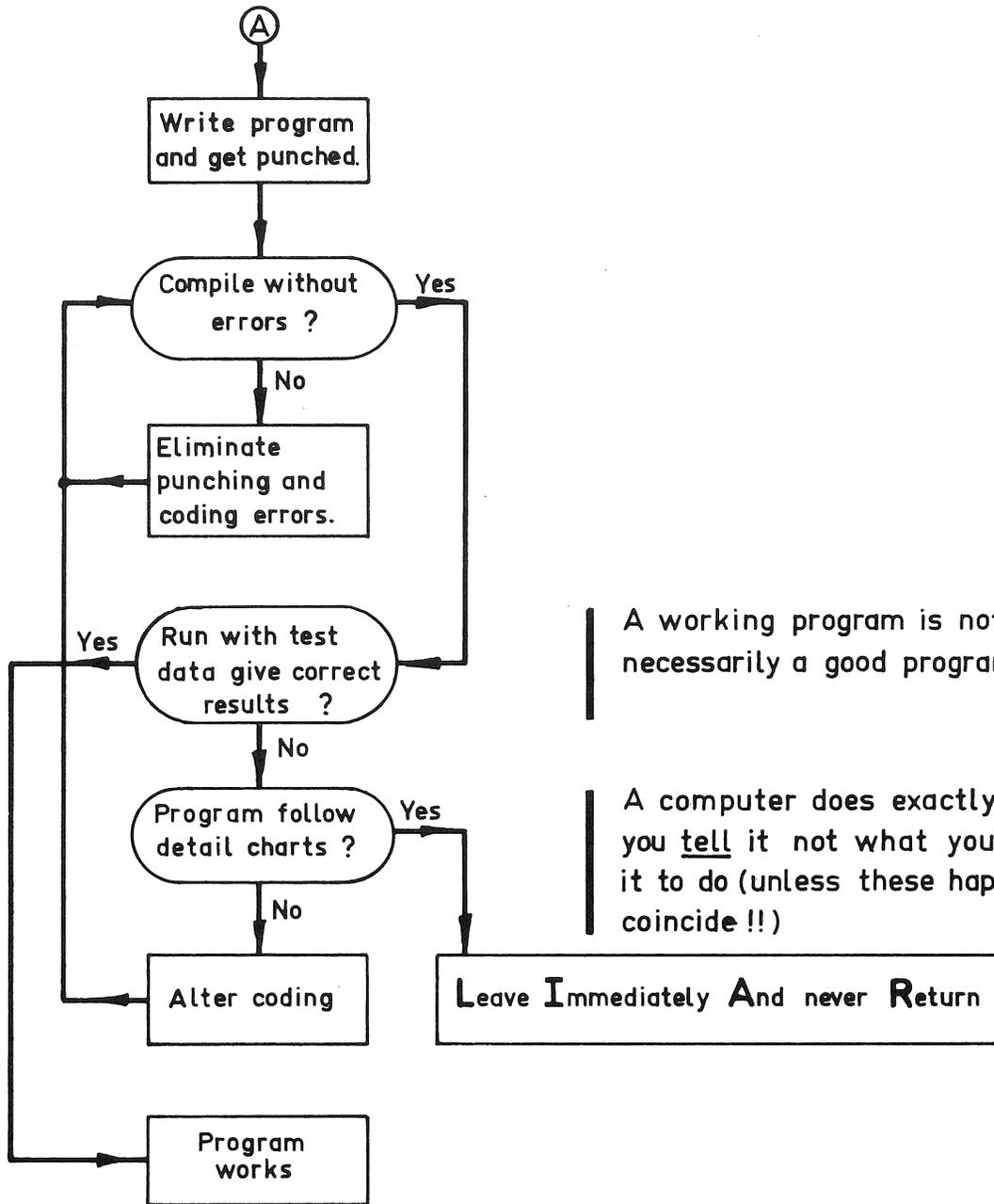
```
┌──────────────────┐
│  Find out what   │
│  is required.    │
└──────────────────┘
        │
┌──────────────────┐
│  Draw system     │
│  chart           │
└──────────────────┘
        │
   ╱ Is this what is ╲     Yes
   ╲ really required? ╱ ──────►
        │ No
┌──────────────────┐
│  Revise system   │
│  chart           │
└──────────────────┘
        │
┌──────────────────┐
│  Draw necessary  │
│  detail charts.  │
└──────────────────┘
        │
   ╱ Dry run with  ╲      Yes
   ╱ test data gives ╲ ──────►
   ╲ correct results? ╱
        │ No
┌──────────────────┐
│  Revise detail   │
│  chart           │
└──────────────────┘
        │
No ◄── ╱ Still satisfied ╲
       ╲ with system  ?   ╱
        │ Yes
Yes ◄── ╱ Satisfied with ╲     No
  (A)   ╲ detail method ?  ╱ ──────►
```

Time spent thinking about the problem and specifying the problem in detail greatly reduces the debugging time and ensures the desired results.

A well documented program can be altered or extended by any competent programmer.

Test data must be simple yet test all options.

(A)

Write program and get punched.

Compile without errors ? — Yes / No

Eliminate punching and coding errors.

Run with test data give correct results ? — Yes / No

Program follow detail charts ? — Yes / No

Alter coding

Program works

A working program is not necessarily a good program.

A computer does exactly what you _tell_ it not what you _want_ it to do (unless these happen to coincide !!)

Leave **I**mmediately **A**nd never **R**eturn

PDP 10 COMPUTER SERVICES

THE COMPUTER CENTRE

The University of Queensland Computer Centre now operates within the newly-created Department of Computer Science, offering computer services to all University Departments and a number of non-University organizations. The Head of the Department of Computer Science (Professor G.A. Rose) is administratively responsible for the operation of the Computer Centre, whilst policy matters affecting computer service and major capital expenditure are determined by the Computer Centre Executive Committee.

THE PDP 10 COMPUTER SYSTEM

The PDP 10* is a dual-processor, multiprogrammed, timesharing computer system, which will augment and extend the computer services currently offered on the GE 225 computer. The introduction of the PDP 10 will result in greatly reduced turnaround on jobs, and a reduction in the costs of processing work.

Initially, batch processing, only, will be available, but the service will gradually be extended to include interactive timesharing from remote terminals, sixteen of which will be installed during 1969.

AVAILABILITY

All University users wishing to use the facilities at the Computer Centre, must first obtain the authorization of the head of their department. (Non-University organizations may also elect to follow a similar procedure). In the case of the PDP 10, this authorization is in the form of an *Order for Computer Services*, obtainable in pad form from the University Store. Each order must be completed in triplicate and two copies forwarded to the Computer Centre before any work is undertaken. One copy will be retained by the Computer Centre, and the second copy will be forwarded to the University Accounts Section. The third copy should be retained by the department concerned for their own records.

---

* A fuller description of the PDP 10 is contained in *Technical Memorandum No. 1 - The University of Queensland Timesharing System*, copies of which may be obtained from the Computer Centre.

Information which must be supplied on each order includes a project number, the department name, the name of the person responsible for the project, a limit of expenditure, and the account to which the expenditure should be charged. The order must, of course, be signed by the head of the department or by some other person authorized to sign orders.

Unlike the system of accounting for the GE 225, each project on the PDP 10 will be uniquely identified by a single number. This *Project Number* will be assigned by the Computer Centre to all new projects, and must be quoted whenever computer services are required. This project number will remain current until terminated by the department, or until the project becomes inactive. A brief description of the work to be carried out is required for all new projects; subsequent orders on existing projects need only quote the project number.

It should be noted that *the limit of expenditure quoted on the order will be interpreted as an absolute limit, and, under no circumstances, will the expenditure be permitted to exceed that amount.*

Subsequent orders may authorize further expenditure against an existing project, and the cost may be charged to any valid account. However, the limit of expenditure on previous orders will be exhausted before any subsequent orders take effect.

USING COMPUTER SERVICES

Unless a computer program already exists for a given job, users will be required to write their own programs. The Computer Centre does not provide a programming service.

Initially, FORTRAN IV will be the only programming language available on the PDP 10, but other languages such as MACRO, AID, BASIC, COBOL, and LISP, will be gradually introduced. From time to time, the Computer Centre will be conducting classes in these languages, and these will be notified as they are scheduled.

Work for batch processing on the PDP 10 should be submitted to the Administrative Officer (Mr. J. Jauncey) or deposited in the input tray in the clients' room. Completed jobs will be stacked under project number, in the output racks, which are also located in the clients' room.

Key punching and verifying services are also available, and should be submitted on standard coding sheets to the Administrative Officer. Two keypunches are available in the clients' room, but the use of these machines is limited to five minutes at a time.

Further details concerning the use of computer services may be obtained from Mr. Jauncey (Ext. 8471).

CHARGES FOR PDP 10 COMPUTER SERVICES

Because the PDP 10 is a timeshared computer system and is able to run several programs concurrently, charges are calculated on the basis of facilities used by a particular program. There are four groups of charges:

1.   Processor Time and Memory Space. A charge is made for the processor time used by each program plus a charge for the core memory required while the program is in execution. This latter charge is calculated from the product of processor time and memory space used.

      Memory space is allocated in the PDP 10 in integral blocks of 1024 words (= 1K words). *The space occupied by any one program is currently limited to 24K words.* However, it should be noted that programs, which occupy a large amount of memory space, will cost more to run and, because of the manner in which the PDP 10 schedules programs for execution, may take a longer time to be completed.

2.   Information Throughput. A charge is made for the quantity of information processed by a program on each input and output file.

      The charge is on the basis of the number of K words (where K = 1024) or part thereof. There are various rates applying to the items of peripheral equipment attached to the PDP 10. These rates are based on the capital costs, the operating costs, and the utilization of the peripheral devices. A charge will also be made for the transfer of information to and from the file storage system.

3.   Storage of Files. Both programs and data files may be stored for future reference. A charge is made on the basis of the size of each file (in K words) and the time for which the information is stored (in days).

4.   <u>Device Setup and Consumable Items</u>. Where any special action is required in setting up a peripheral device, e.g. to allow special paper to be used, a device setup charge will be made. A fee will also be charged for access to a file which has been preserved on behalf of a user. Access to system files will be allowed free of charge.

Where the use of consumable items, such as printer paper or plotter paper, is excessive, these items will be charge for.

All charged will be calculated as a run is processed. Upon completion of a job, the charge so far accumulated for each item will be rounded to the nearest cent, and the total of all charges for that job debited against the authorized expenditure for that project. A schedule of charges on the PDP 10 for University Departments or for Non-University Organizations is available from the Computer Centre.

INVOICES AND STATEMENTS

Each month, a *Statement of Account* will be forwarded to each University Department. For each project, this will contain details of the total expenditure this month, the total expenditure authorized so far this year, the total expenditure so far incurred, and the balance still authorized. These totals will be summarized for each departmental account, and for the department as a whole. The Accounts Section will use this information to automatically debit the appropriate departmental accounts.

Non-University Organizations will be invoiced on a monthly basis. The expenditure against each project will be itemized against the appropriate order.

LIBRARY ACCESSIONS

This section details the books and periodicals relating to the computer field, that have been acquired by Libraries of the University of Queensland in November and December, 1968.

November Accessions

Stewart, David J. ed. *Automation Theory and Learning Systems*. 1967.
      (001.53 STE, Main Lib.)
Machine Intelligence. No. 1; 1967 (001.535 MAC, Engin. Lib.)

Practical Problems of Library Automation. 1967. (Q029.7 PRA, Main Lib.)

*Interdisciplinary Clinic on Instrumentation Requirements for Traffic Control
    Systems.* New York, 1963. Traffic Control 1965. (388.31 INT, Engin. Lib.)

Clyde, Dean J. *Multivariate Statistical Programs.* 1966. (Q519.92 CLY, Engin. Lib.)

Coleman, Monte W. *BRLESC Plotting Subroutines.* 1967. (Q519.92 COL, Engin. Lib.)

European Atomic Energy Community. Scientific Data Processing Centre. 1965.
    (Q519.92 EUR, Engin. Lib.)

Hockney, Roger W. *Minimum Multiplication Fourier Analysis.* 1965. (Q517.355 HOC,
    Engin. Lib.)

Knuth, Donald E. *The Art of Computer Programming.* [C 1968] (519.92 KNU,
    Engin. Lib.)

Mueller, Dennis J. *Fortran Subroutines for Householder's Method in the Complex
    Case and Eigen Systems of Hermitian Matrices.* 1966. (Q519.92 MUE,
    Engin. Lib.)

Seminar on Linear Programming in Practice. University of New South Wales, 1960.
    Proceedings 1960. (Q519.92 SEM, Maths. Lib.)

Zukhovitskii, Semen I. *Linear and Convex Programming.* 1966. (519.92 ZUK,
    Engin. Lib.)

Computer Digest. V. 3; 1968 [and onwards] (510.78 COM, Engin. Lib.)

Journal of Computer and System Sciences. V. 2; 1968 [and onwards] (510.78 JOU,
    Engin. Lib.)

Moseley, Maboth. *Irascible Genius: a Life of Charles Babbage.* 1964. (608.742
    BAB, 3 MOS, Main Lib.)

Holmes, James F. *Data Transmission and Data Processing Dictionary.* 1965. (621.3803
    HOL, Engin. Lib.)

Baker, C.L. *JOSS: Console Design.* 1967 (Q651.89 BAK, Engin. Lib.)

Bryan, G.E. *JOSS: Accounting and Performance Measurement.* 1967 (Q651.89
    BRY, Engin. Lib.)

Coffman, Edward G. *Stochastic Models of Multiple and Time-Shared Computer
    Operations.* 1966. (Q651.84 COF, Engin. Lib.)

Evarts, Harry F. *Introduction to PERT.* 1964. (658.5 EVA, Main Lib.)

Gonzalez, R. *Associative Techniques for Control Functions in a Multi-Processor
    Simulation Investigation.* 1967. (Q651.89 GON, Engin. Lib.)

Greenwald, I.D. *JOSS: Disc File System.* 1967. (Q651.89 GRE, Engin. Lib.)

Institution of Cost and Works Accountants. *Management Information Systems and
    the Computer.* 1967 - (658.505 INS, Main Lib.)

Kintner, Paul M. *Electronic Digital Techniques*. 1968. (651.8 KIN, Engin. Lib.)

McCrimmon, Kenneth R. *An Analytical Study of the PERT Assumptions*. 1962. (Q658.5 MACC, Main Lib.)

Massachusetts Institute of Technology. *Project MAC: Progress Report III*. 1966. (Q651.8 MAS, Engin. Lib.)

Rohrbacher, Donald L. *Advanced Computer Organization Study*. 1966. (Q651.89 ROH, Engin. Lib.)

Totschek, Robert A. *An Empirical Investigation into the Behaviour of the SDC Time-Sharing System*. 1965. (Q651.84 TOT, Engin. Lib.)

Grench, R.E., comp. *Collected Algorithms*. 1960-1963. 1965 (Q510.78 GRE, Engin. Lib.)

<u>December Accessions</u>

Computers and the Lawyer. 1968. (Q510.7834 COM, Law Lib.)

Morris, Stanley M. *Salem - A Programming System for the Simulation of Systems Described by Partial Differential Equations*. 1968. (519.92 MOR, Engin. Lib.)

The Computer Bulletin. V. 12, No. 4; 1968, and onwards. (510.78 COM, Engin. Lib.)

Symposium on the Prospects for Simulation and Simulators of Dynamic Systems, Baltimore, 1966. Prospects for Simulation and Simulators of Dynamic Systems. 1967. (610.7 SYM, Engin. Lib.)

Saxon, James A. *Basic Principles of Data Processing*. 1967. (651.8 SAX, Engin. Lib.)

<u>FORTRAN MEMORY ALLOCATION ON THE PDP 10</u>

A question frequently put to the Computer Centre staff is: how large a FORTRAN program will I be able to run on the PDP 10? There is no brief answer to this question, and, in particular, it is not possible to base an estimate of the amount of core needed on the PDP 10 upon a knowledge of the amount of core used by the same program when run on the GE 225. This article contrasts the methods of organizing the memory in the PDP 10 and GE 225 FORTRAN IV systems. An appreciation of the different philosophies involved
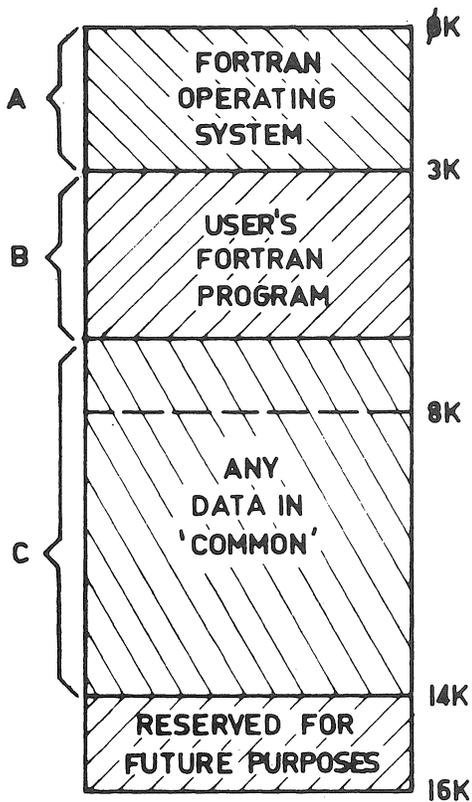
FIGURE.1. Memory allocation in the
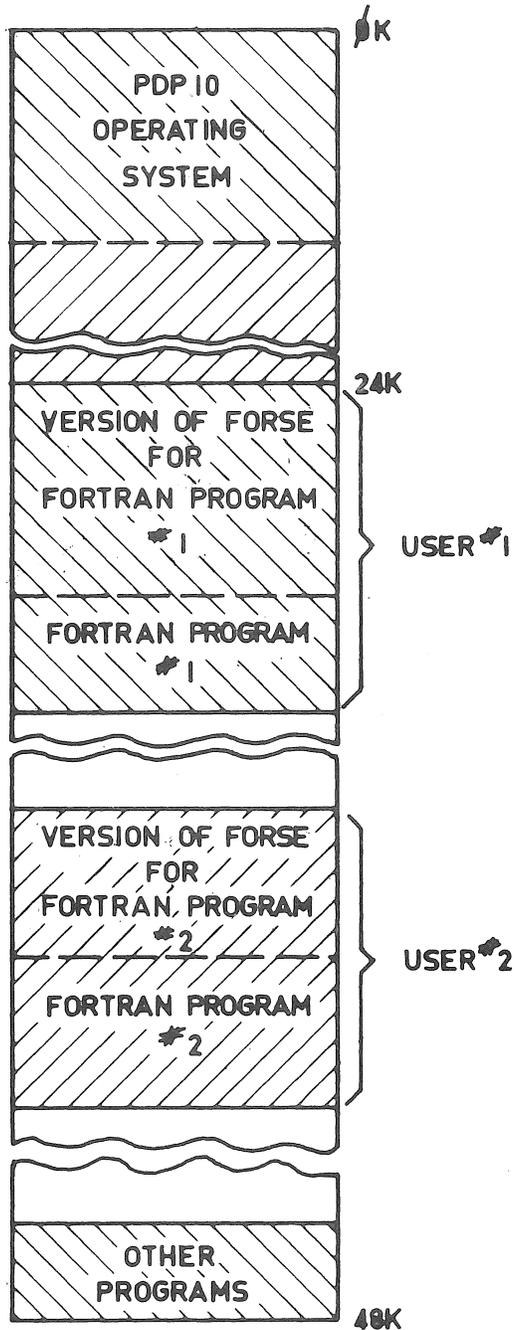GE-225 during a FORTRAN IV
run.

FIGURE.2. Typical Memory
allocation in the PDP-10.
Two FORTRAN users
are shown in core
together.

will enable the FORTRAN programmer accustomed to the GE 225 to design programs which make optimal use of core space on the PDP 10.

Let us begin by reviewing the situation for the GE 225. Here, only one program can be stored in the memory at a time, so that more than half the memory can be left unused for quite long periods. The first 3K (where $K = 2^{10} = 1024$) memory locations are occupied by the GE 225 FORTRAN IV Operating System (region A in Figure 1). The next 5K locations are available for the user's FORTRAN program (which is shaded as region B in Figure 1), and the remainder (region C) may be used for the storage of data in COMMON storage. Object program instructions and simple variables cannot be stored in the upper half of the memory.

The FORTRAN IV Operating System for the GE 225 is of fixed size and consists of a large number of implicitly-called subroutines for performing input/output, conversion of numbers, etc. It contains subroutines to handle F, E, I, O type format specifications (among others), even though few programs use all of these. Reference in a program to a mathematical function such as COS or SQRT, however, automatically causes the appropriate subroutine to be placed in region B, thereby expanding that region. The only *mathematical* subroutines loaded into core are those actually used. (A few implicitly-called input/output routines are also handled in this way).

With these aspects of the GE 225 system in mind, let us consider the PDP 10. Figure 2 provides a typical "snapshot" of core at a time when two FORTRAN programs are in core together, a situation which cannot exist on the GE 225. The first 16K memory locations are reserved for the PDP 10 Operating System, which is permanently resident in core. It is common to all programs being executed, and performs functions such as input/output, time accounting and scheduling. At any given instant, the remaining core may be occupied by a single program or by several small programs*. As shown in Figure 2, each FORTRAN program is loaded with its own tailormade version of FORSE which acts as an interface to the Operating System.

---

* While this would imply that a single program could occupy the remaining 32K of core, in fact the operating system limits all programs to a maximum size of 24K. This is done to permit reasonable efficiency in the scheduling of jobs for execution, and also to ensure a satisfactory response time for all jobs.

FORSE is similar to the FORTRAN operating system on the GE 225, in that it contains routines falling into two main categories -"mathematical" and "input/output". In the case of the input/output routines, however, two significant differences appear. Firstly, since the Operating System is responsible for input/output operations on the PDP 10, these routines are smaller than otherwise, as they have only formatting tasks to perform. Secondly, only those input/output routines actually required are loaded.

It can be expected, then, that the size of FORSE for a particular job will depend greatly on the variety of features used in the program. Programs that read in FORMAT statements at run-time will have the largest Operating Systems, since there is the possibility of any type of format conversion (E, I, F, etc.) being found on a data record. However, even in this instance, FORSE will normally occupy considerably less memory space than on the GE 225.

Other factors must be considered when comparing the use of memory space on the PDP 10 and the GE 225. For programs which contain large arrays (REAL, COMPLEX, or DOUBLE PRECISION) an important consideration is that each array occupies only half as many words on the PDP 10 as on the GE 225. Therefore, floating point numbers occupy one word in the PDP 10 (28-bit mantissa and 8-bit exponent), but two words on the GE 225 (31-bit mantissa and 9-bit exponent). Similar considerations apply to integer and character representations.

On the PDP 10, a FORTRAN program occupies an exact number of 1K blocks of core memory (and the charge for the use of memory space is varied correspondingly). This area of core may be arbitrarily subdivided into "program" and "COMMON" areas. There is no restriction, such as exists for the GE 225, that the "program" part be confined to remain wholly within the lower half of the area available.

Finally, the powerful hardware instruction set of the PDP 10, combined with an efficient optimization process in the FORTRAN compiler, gives a further significant reduction in the memory space required by a program. While not necessarily a typical example, a randomly-chosen FORTRAN IV program which occupied 8314 memory locations on the GE 225 was found to require only 4321 locations on the PDP 10. (Each figure includes the appropriate pre-coded routines).