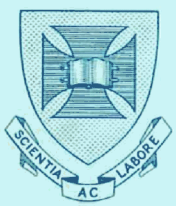
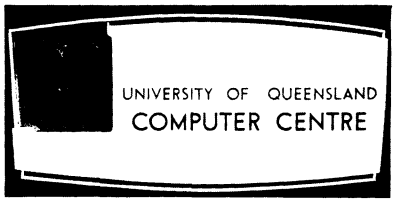


UNIVERSITY OF QUEENSLAND

COMPUTER CENTRE

Information Manual



FOREWORD

As a result of generous support from State Government Departments, Local Authority organisations and private industry, a computer centre operated on a co-operative basis has been established in the University of Queensland.

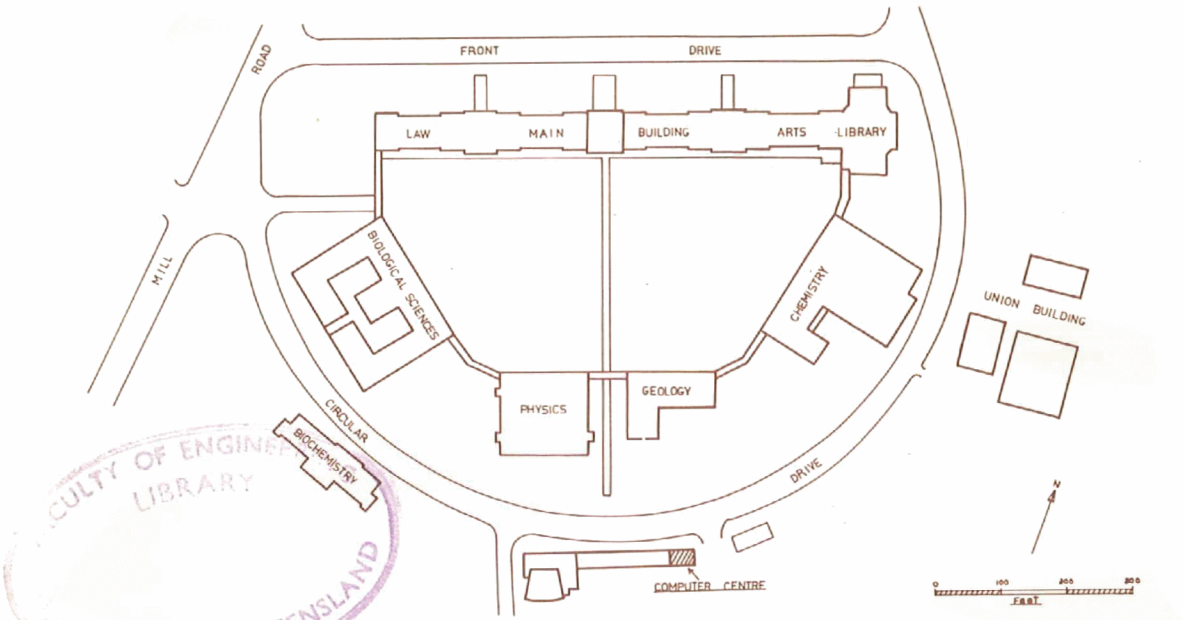
After calling tenders for the computer installation, the University placed an order for a GE 225 digital computer, manufactured by the General Electric Co., U.S.A., and this machine was delivered in March, 1962. The computer is housed in an air-conditioned section of the new main Engineering building at St. Lucia, as illustrated.

To assist users of the computer centre, a technical description of the equipment, including programming, has been compiled from information supplied by Australian General Electric Pty. Ltd. Except for the description of the ancillary equipment, most of the data and illustrations have been extracted from the Reference Documents listed under the Table of Contents. The section describing peripheral equipment indicates possible future expansion of the computer facilities, but these items have not been included in the initial installation.

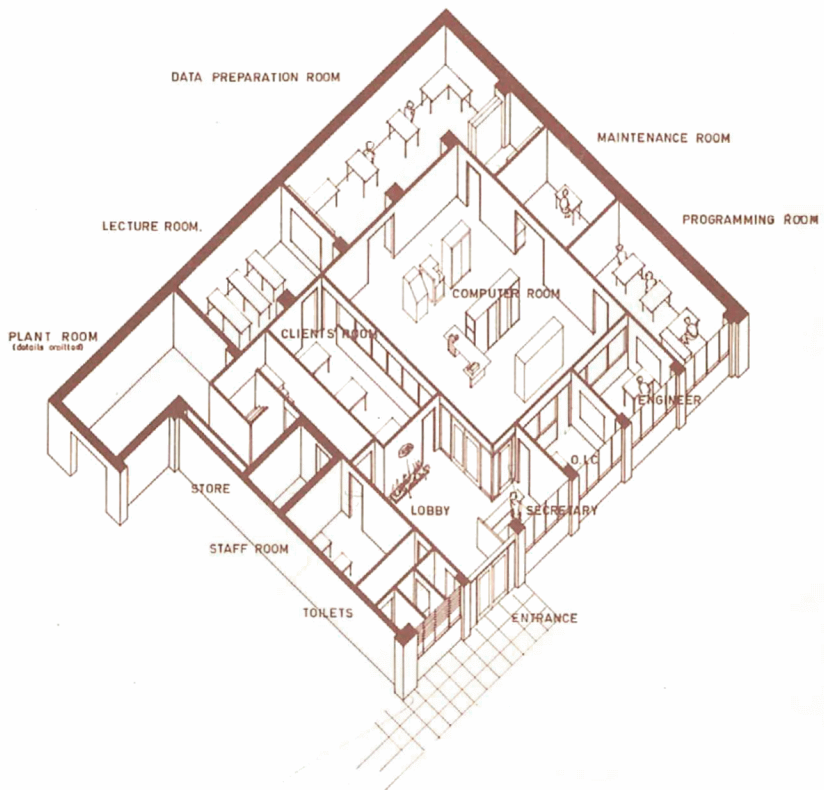
S. Lawrence

Chairman, Computer Centre Advisory Committee

February, 1963.



LOCATION OF COMPUTER CENTRE - UNIVERSITY OF QUEENSLAND, ST. LUCIA



ISOMETRIC SKETCH OF COMPUTER CENTRE

TABLE OF CONTENTS

REFERENCE DOCUMENTS

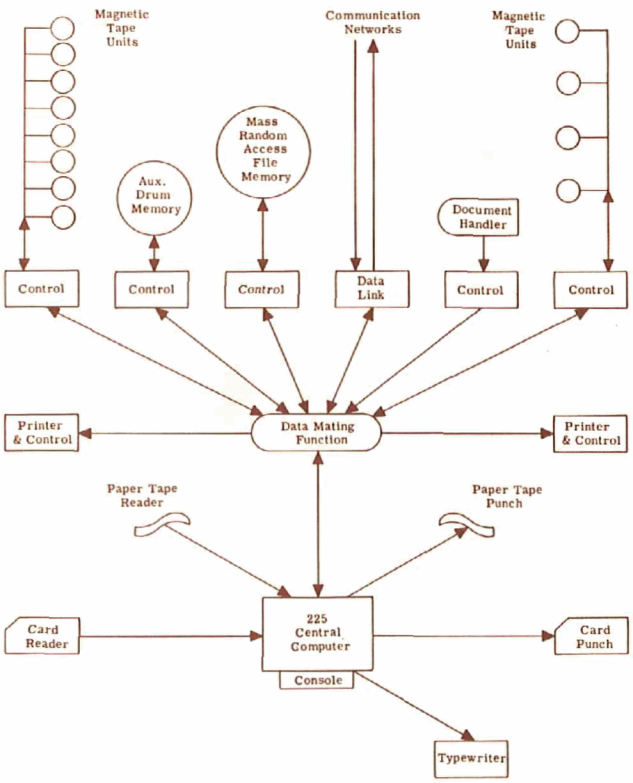
1. GE 225 Systems Manual - G.E. Publication CPB - 98
2. GE 225 Programming Manual - G.E. Publication CPB - 126A
3. General Compiler Manual - G.E. Publication CPB - 123
4. GE 225 WIZ System - General Electric Computer Dept.
5. GE 225 WIZOR System Reference Manual CD225H4.00 - General Electric Computer Dept.
6. GE 225 ZOOM - A Macro Assembler - CD225F1.002 - General Electric Computer Dept.

PART I
EQUIPMENT

1. GENERAL DESCRIPTION

The major equipment installed in the computer centre is a GE 225 stored program, general purpose, digital computer. The basic computer, which is fully transistorised, consists of a *Central Processor* with a control console and typewriter. Both paper tape and punched card input/output are provided together with an auxiliary arithmetic unit and a high speed printer.

It is, however, intended that the installation can be expanded to meet growing demands; optional peripheral equipment offered by the manufacturer includes high speed magnetic tape units, additional high speed printers, mass random access memories, and magnetic document handlers. These various peripheral devices communicate with the Central Processor through the "data mating function", which is a common control and transfer point for the units. This arrangement allows for the addition of peripheral equipment as the needs of the installation grow, as well as for the addition of new or improved devices to the system, and allows the Central Processor to time-share associated peripheral devices with internal computations on an automatic basis. After a peripheral control unit has been selected and has received all the necessary instructions from memory, the peripheral unit operates on a semi-off-line basis. Except for the time periods during which the peripheral unit interrupts the central computer to enter information into memory or to read information from memory, the peripheral unit operates independently of the computer.



Large Configuration

The diagram illustrates one possible configuration of a large system. In this arrangement, the typewriter, card reader, card punch, paper tape reader, and paper tape punch tie in directly to the central computer, while the peripheral units are connected to the central computer through the data mating function.

2. AUTOMATIC PROGRAM INTERRUPT

The central processing unit may be fitted with a Program Interrupt feature which provides for automatic interrupt of the main program whenever a peripheral controller calls for access to the central processor. This allows a degree of multi-program operation such that a program with little calculation but large input-output requirements may be time shared with a program of the opposite type, or other programs using different peripheral equipment.

3. WORD FORMAT

The computer operates arithmetically in the binary mode. Information exists in the computer in the form of *words*, each word consisting of 20 binary digits (bits). A word stored in memory has 21 bits since a parity bit is computed as the transfer to memory occurs.

3.1 Data Word Format

A *binary data word* consists of 19 binary bits plus a sign bit. Bit 0 is the sign bit; 0 is a plus and 1 is a minus. For example, the decimal number +49 is represented as follows:-

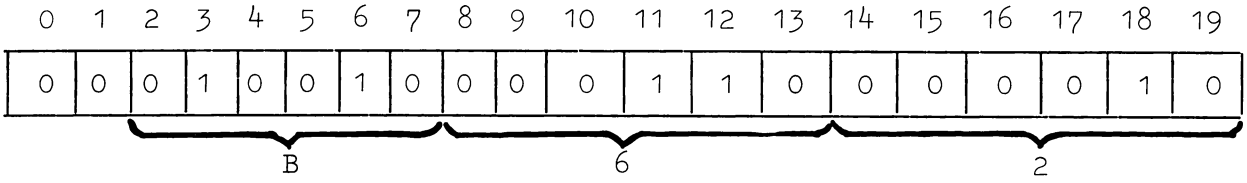
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1

Special instructions are provided to handle computations on double length data words; that is, 38 bits plus the sign. A 20-bit binary word will accommodate a decimal numerical value of plus or minus 524, 287; a 38-bit word covers the range plus or minus 274, 877, 906, 943.

A *numeric data word* may, therefore, be most conveniently regarded as consisting of approximately $5\frac{1}{2}$ decimal digits. Double length operations permit the handling of numbers of up to 11 decimal digits. Subroutine packages provide for the conversion of decimal numbers to binary notation.

Information is entered into the computer by reading punched cards or punched paper tape. The information may be punched in 80 column alphanumeric (Hollerith) code, in binary (column) form, or in alphanumeric teleprinter code. Thus, the computer can function as an alphanumeric machine with binary capabilities or as a normal binary computer, and the user may switch between modes of operation to take advantage of the particular characteristics of a given application.

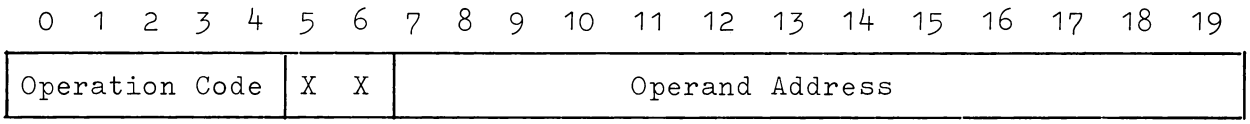
Upon reading a card or tape in the alphanumeric mode, each character is converted into a six-bit binary coded decimal configuration. Thus, three alphanumeric characters will occupy 18 of the 20 bit positions of a memory location. In this sense, an *alphanumeric data word* consists of three alphanumeric characters. Double length operations also permit the automatic handling of symbolic codes (account numbers, stock numbers, etc.) of six alphanumeric characters. As an example, the symbolic code B62 in binary coded decimal form would appear as follows:-



Bit positions 2 to 19 store the three six-bit binary coded decimal digits, while bit positions 0 and 1 may be used to hold other information.

3.2 Instruction Word Format

An *instruction word* in the computer is a single address word consisting of 20 bits. Except for branching operations, instructions are executed sequentially. The reading of the next instruction from memory occurs after the execution of the current instruction. A sequence control counter, the P counter, contains the address of the next instruction to be executed. The basic format of the instruction word is as follows:-



Bits 0 to 4 designate the operation which is to be performed, bits 5 and 6 determine whether or not the instruction address is to be automatically modified, and bits 7 to 19 indicate the operand address.

Since the five bit positions allowed for the operation code can define a maximum of 32 operations, or commands, it is clear that more bit positions are required to define any further operations. This is achieved through the use of bit positions in the operand address field for those instructions which require only a limited portion of the field (e.g., shift commands require only bit positions 15 to 19 to indicate length of shift) and for those instructions which do not have an operand address (e.g., word transfers between registers).

Bits 5 and 6 of an instruction word indicate whether the instruction address is to be automatically modified, before execution, by the selection of one of three index registers and the addition of its contents to the operand address portion of the instruction word. These three registers are memory locations 1, 2 and 3. If an instruction in the I (instruction) register calls for automatic address modification, an extra word time (18 microseconds) is required to accomplish this.

4. THE CENTRAL PROCESSOR

4.1 Memory

The *memory* of the computer is made up of magnetic cores, each core storing one bit of information. The core memory provides storage for both instructions and data, having a storage capacity of 8,192 words. This can later be expanded to 16,384 words. A word exists in memory as 21 binary digits; 19 information bits, sign, and a parity bit. Each word in memory is individually addressable with random access, and can be addressed, transferred to the memory register, and regenerated in its original location in one 18 microsecond cycle.

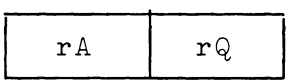
When a word is transferred to memory, a parity bit is computed and stored along with the rest of the word. When a word is transferred from memory, a parity bit is again computed, and this newly computed parity bit is compared with the parity bit already existing in memory. This checks against the loss of information during transfer and storage in memory. Data transfers to and from peripheral equipment are also checked.

4.2 Arithmetic and Control Registers

The *arithmetic and control registers* form the "heart" of the computer where the calculation and control functions are performed.

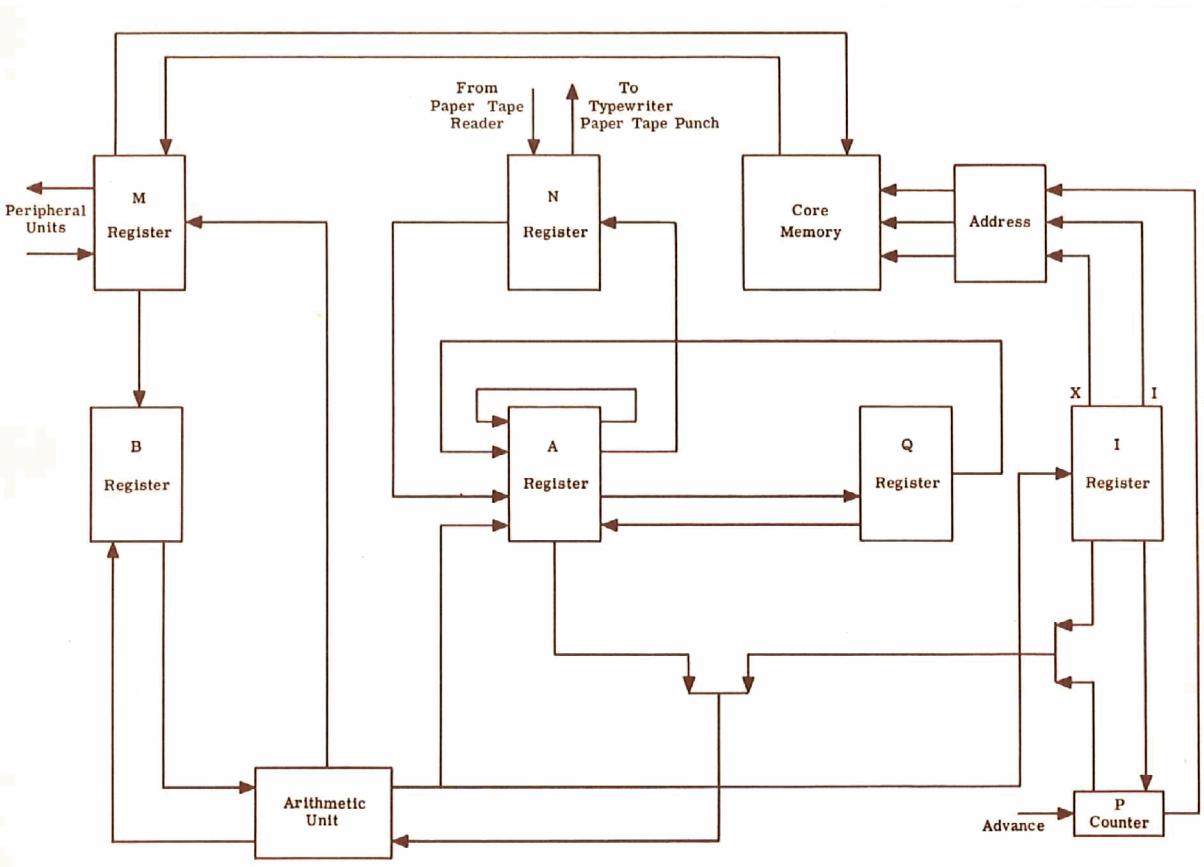
The *A register* is a 20-bit register which serves as the accumulator for the central processor.

The *Q register* is a 20-bit register which acts with the A register to form a 38-bit (plus sign) accumulator during the execution of double length instructions. Information is not transferred directly from memory into the Q register but is transferred through the A register into the Q register.



Q holds the least significant half of a double length word read from memory. After multiplication, Q holds the least significant half of the product. During division, Q holds the least significant half of the dividend; after division, Q holds the remainder.

The *N register* is a six-bit register which is used as a single character buffer between the computer and the typewriter, paper tape reader or paper tape punch. Information is transferred directly between the N register and the A register.



Register Relationships

The *I register* is the instruction register. It holds the 20 bits of the instruction word during the execution of a computer command.

The *P counter* is the location counter which controls the execution of the instructions; that is, it holds the memory address of the next instruction to be executed. The P Counter is incremented by one before the execution of an instruction so that it normally indicates the address of the next instruction in sequence. The contents of the P Counter can be set from the I register.

The *B register* is a 20-bit register which acts as a buffer register during data transfers, as a buffer for arithmetic operations, and is used in the execution of certain data transfer commands.

The *M register* is a 21-bit register which acts as a buffer between the magnetic core memory and the B register and also acts as a buffer to

peripheral devices. The 21 bits in the M register include 20 information bits plus a parity bit.

The *arithmetic unit* serves two functions. During arithmetic operations, it performs the arithmetic calculations specified by the command code in the I register. It also serves as a transfer bus for data words going to the A register and for instruction words going to the I register.

4.3 Control Console

The primary function of the *control console* is to provide an indicating and control centre for the computer operator from which he has visual representation and manual control of the operation of the system. The indicator panel displays the contents of the A register, or accumulator; the I (instruction) register; and the P counter. Various alarm lights and twenty external control or breakpoint switches are provided, together with other controls such as Manual Operation, Data Transfer, Power On and Power Off. Registers and memory cells may be loaded or examined, using the console controls.

4.4 Console Typewriter

The *console typewriter* is an on-line output and control device. It is similar to an ordinary electric typewriter, but with additional functions. The typewriter can type at a rate of 10 alphanumeric characters per second to monitor system operations, prepare short summary reports, give operator instructions, etc. Computation and operation of other peripheral devices may proceed while the typewriter is in use.

5. AUXILIARY ARITHMETIC UNIT

The ***Auxiliary Arithmetic Unit*** is a single address binary arithmetic unit operating as a "slave" to the Central Processor. It enables the computer to perform floating point operations automatically, virtually eliminating the necessity for problem scaling. In this respect, programming is simplified and the calculating speed of the computer is increased. In the computer, a floating point number consists of two 20-bit words - the format being 30 bits for mantissa, 8 bits for exponent and two sign bits. Floating point operations may be performed in normalized or unnormalized modes and commands may be indexed in the usual manner.

The unit consists of two additional 40-bit registers (AX and QX) a one word buffer (MX), and control circuitry for multiplication and division of double length words in fixed point mode, and for automatic floating point operations. This unit connects to the Central Processor through the data mating controller, and will operate simultaneously with the Central Processor on the various peripherals.

6. INPUT/OUTPUT EQUIPMENT

6.1 General

The computer has available both punched paper tape and punched card input/output devices. Although paper tape is a cheaper medium and is particularly suited to scientific applications, punched card input/output has been provided for those wishing to use the unit record approach of punched cards.

6.2 Paper Tape Reader and Punch

The ***paper tape reader*** is an on-line device which reads alphanumeric information from punched paper tape at a speed of either 250 or 1000 characters per second. Standard paper tape used in the Computer Centre is 5 channel in $^{11}/_{16}$ inch width, although the reader will accept 6 and 7 channel in $^{7}/_{8}$ inch width or 8 channel in 1 inch width. The punch density is 10 characters per inch.

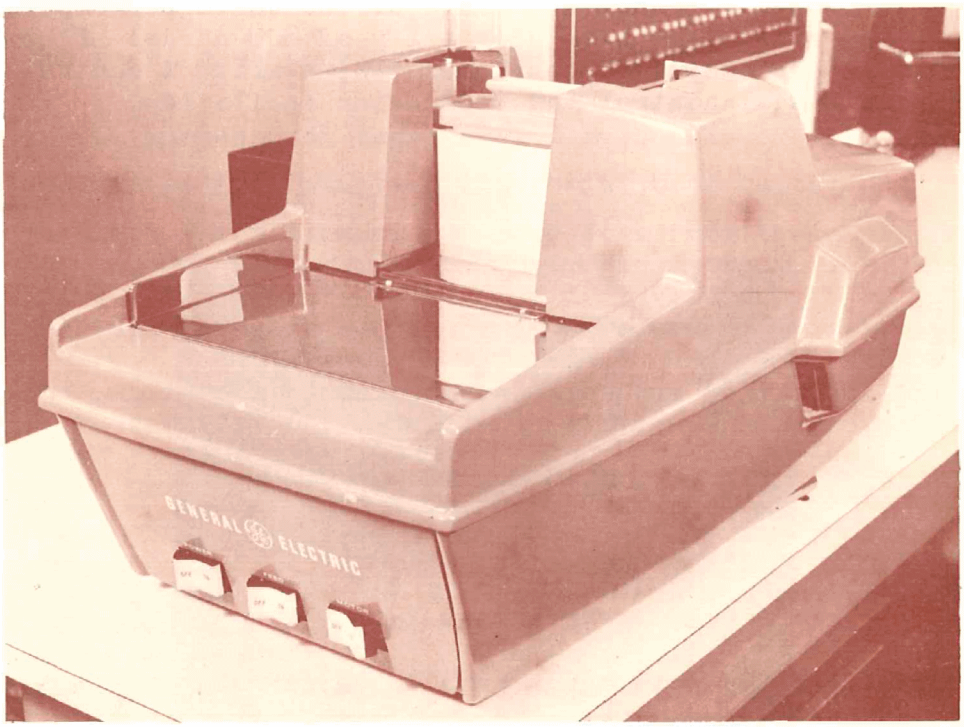
At the lower tape speed, the reader will stop on the character being read; at 1000 characters per second, it will stop between characters. Checking features include a media presence check and an end of tape indicator. Data rearrangement is accomplished by computer program.

The ***paper tape punch*** is an on-line output device which punches alphanumeric information into paper tape at a speed of 110 characters per second. Paper tape 5 channel 11/16 inch width, 6 and 7 channel on 7/8 inch width, or 8 channel on 1 inch width may be punched.

Checking features include a media presence check and parity generation. Data rearrangement is accomplished by the computer program.

6.3 Card Reader

The card input equipment consists of a *card reader* and a set of input logic circuits which link the card reader to the computer. The card reader reads standard $7\frac{3}{8}$" x $3\frac{1}{4}$" 80-column, 12-row cards. Cards are read column-by-column and may be fed card-by-card or continuously at the rate of 400 cards per minute. The input and output magazines each have a capacity of 600 cards. Provision is made for both manual and computer programmed stop-start. Checking features include synchronization and optical equipment checks.



Three card formats may be used with this reader:

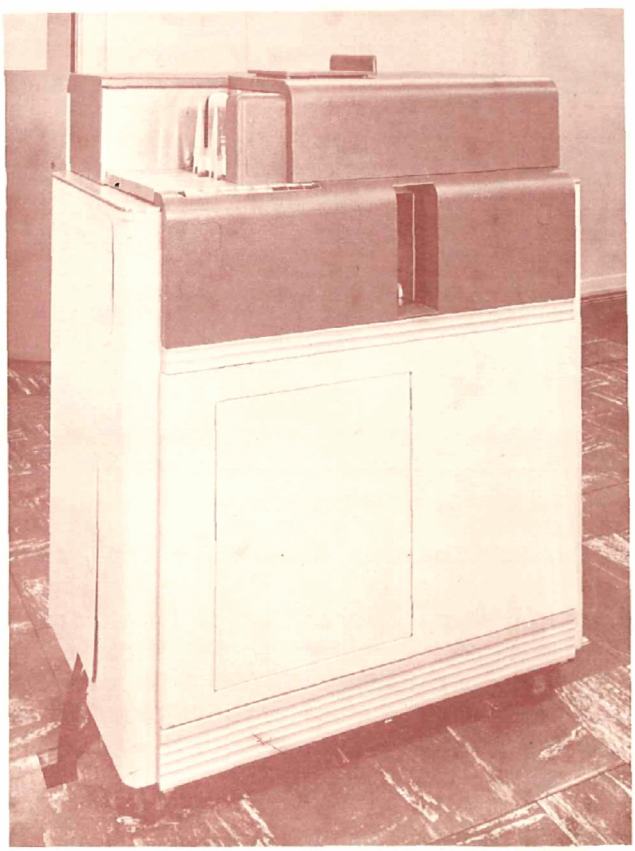
1. The Alphanumeric Card Format is one character per card dolumn, 80 characters per card, punched in Hollerith code. The Hollerith code and the corresponding GE 225 bit configurations are given in Appendix I. Acceptable characters include the 26 letters of the alphabet, numbers 0 to 9, and twelve special characters and punctuation marks.

2. The Binary Card Format is 10 bits per card column or one half of the 20-bit GE 225 word, i.e., 40 words per card. The first card column of any two columns comprising a full word contains the most significant bits of the word. In the binary mode, rows 11 and 12 of the punched card are not used.

3. The card reader will also read binary card formats of twelve bits per card column. In this mode, called the full card mode, each column is sent to one GE 225 word so that 80 machine words are required in this case each receiving twelve bits.

6.4 Card Punch

The card output equipment consists of a *card punch* and a set of output logic circuits which link the card punch to the computer. The card punch punches standard Hollerith 80-column cards. Information is punched row-by-row on a card-by-card basis at a rate of 100 cards per minute. The input and output magazines each have a capacity of 800 cards. Provision is made for both manual and computer programmed stop-start. A control panel on the punch furnishes manual control and indication facilities.



The card punch equipment can punch cards in the two formats described in the previous section on the card reading equipment. Data re-arrangement can be achieved through program and/or card punch plugboard control.

7. ANCILLARY EQUIPMENT

7.1 General

Ancillary equipment for the preparation of paper tape and punched cards is available to users of the computer. An extensive range of both types of equipment has been installed and the user may select the mode of data preparation which best suits his own requirements. Data preparation is carried out on these equipments by skilled staff thus removing the necessity of such work from the user.

7.2 Paper Tape Equipment

The facilities provided by the paper tape equipment include the following:-

(1) manual punching of tapes from a keyboard;

(2) facilities for detecting errors in these tapes;

(3) facilities for producing a tape free from these errors;

(4) means for copying tapes;

(5) facilities for printing out tapes.

The paper tape may be punched manually with a teleprinter or with a separate keyboard perforator. When punching is carried out on the teleprinter, a print-out is also obtained giving a visual check on the accuracy of punching. Many errors are detected during punching and may be corrected immediately. Other errors may be detected by punching two tapes separately and comparing them automatically or by proof-reading the print-out. A tape reader may be connected to the teleprinter for producing a tape free from errors, as a means of copying tapes, or for printing out tapes.

At the present time the Computer Centre has available two keyboard perforators, two interpreter/reproducer sets, one verifier and one comparator, and additional equipment may be added as required.

7.3 Punched Card Equipment

The preparation and editing of punched cards involve the operations of punching, verifying, sorting, reproducing and tabulating. The normal method of producing cards free from error is by performing two punching operations on the *punch* and *verifier* . Any differences between the two operations are indicated on the verifier. Sorting is necessary to ensure that cards are in order and a deck of cards may be copied on the *reproducer.* Print-out from cards is usually performed on a *tabulator* although it may be obtained with a card-operated typewriter.

At the present time the Computer Centre has available two card punches and one verifier with access to sorting and reproducing equipment. Additional equipments may be added from time to time as need arises.

7.4 Supplementary Equipment

To assist in carrying out computations to check on computer results, a fully automatic calculating machine is provided. This *calculator* has an accuracy comparable with that of the computer. In addition, an *adding machine* with a paper tape perforating attachment is available for punching data and, at the same time, producing a check sum.

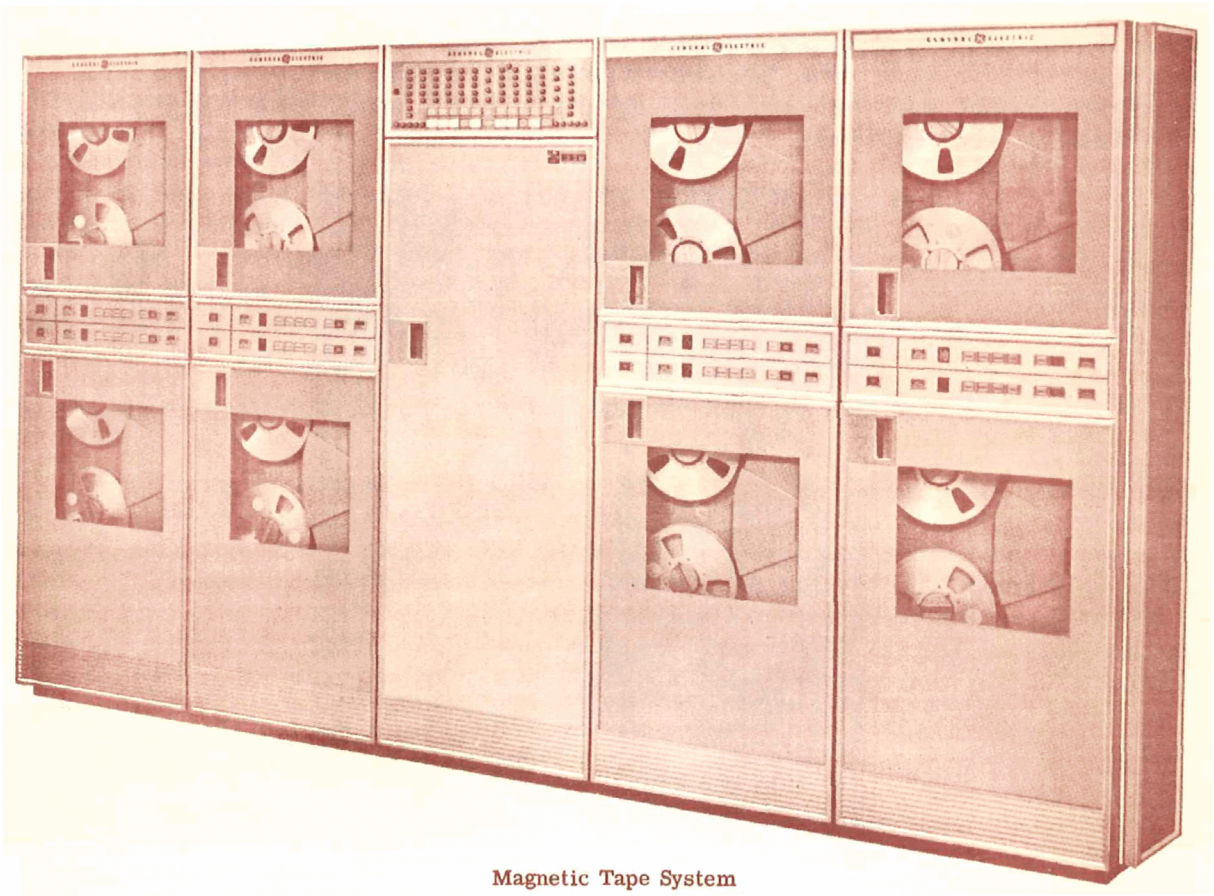
8. PERIPHERAL EQUIPMENT AND MACHINE LOGIC

8.1 General

As outlined in Section I, various items of peripheral equipment may be connected to the Central Processor through the data mating link, and the following items are described hereunder for the purpose of completeness. *The high speed printer is the only one of these items at present available with the system installed by the University.*

8.2 Magnetic Tape System

Magnetic tape is required for sequential storage of large files of information. One or more *magnetic tape handlers* can be connected to the Central Processor. These tape units operate with data transfer rates of 15,000 characters per second, and may read or write concurrently with the operation of other peripheral equipment and the Central Processor.



Magnetic Tape System

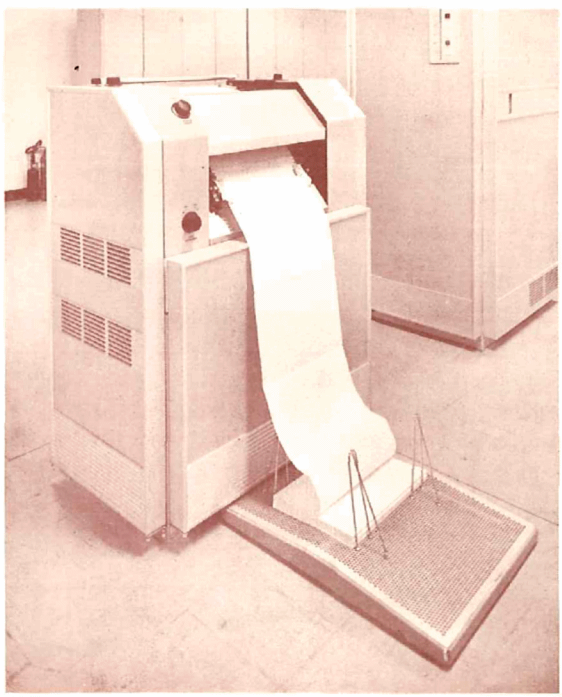
Information is recorded on magnetic tape in groups of words called "records". The minimum record length is one word; the maximum record length is 16,384 words. Information is recorded at a density of 200 characters per linear inch, a total of over 8 million characters per reel of tape.

On-line control of the tape units is achieved by the computer program, while off-line control is available through a small operator panel associated with each tape unit.

The tape control unit reads or writes magnetic tape in two distinct modes: binary or six-bit binary coded decimal. The mode of operation of the tape control unit is determined by the programmed command. Information recorded on GE 225 magnetic tape is compatible with other computer systems now in use. Appendix I gives additional technical information on this point. The tape system has error detection circuits to ensure accuracy in transferring information between memory and tape. Both lateral and horizontal parity bits are recorded on tape and parity checks are applied during reading and writing.

8.3 High Speed Printer

The *high speed printer* is used for the printing of reports and statements and for the large scale output of information. The printer produces alphanumeric output, up to 120 positions per line, at the rate of 600 lines or 900 lines per minute. The printer operates concurrently with other peripheral equipment and the processing unit, utilizing less than 3% of the available memory access time.



There are a total of 50 printable characters which can be represented: numbers 0 to 9, the 26 letters of the alphabet, and 14 special characters.

8.4 Magnetic Document Handler

The *magnetic document handler* is an on-line or off-line device which reads and sorts, at a speed of 1200 items per minute, paper documents printed with E13B font and magnetic ink. These documents may vary in quality, size and degree of mutilation. Operation proceeds concurrently with internal computations and the operation of other peripheral units.

8.5 Mass Random Access File

The *mass random access file* is an on-line, large capacity, file storage device consisting of several large magnetic discs arranged vertically on a rotating shaft. Over six million words of information are randomly accessible with this device. From one to four memory units may be connected to the Central Processor through a *controller*.

8.6 The Automatic Program Interrupt Feature

Although not strictly an item of peripheral equipment, the *automatic program interrupt* hardware is available as an attachment to the Central Processor. It allows an interruption of any function of the computer at any time to perform other operations. All pertinent registers containing data from the main program are stored automatically and, at the end of the interrupt, restored so that the main program may continue from where it was interrupted. Program interrupt operations may continue unaffected during all operations of the Central Processor including set-up, program debugging, and normal program operation.

8.7 Three way Compare: 96 Index Registers: Decimal Add and Subtract

These facilities can be provided as an additional attachment to the Central Processor. The *three way compare* facility enables two quantities to be compared and the decision - "greater than", "less than", or "equal to" - obtained. Three index registers are standard equipment on the GE 225 but this may be increased to a total of *96 index registers* , giving the programmer increased flexibility. The *decimal add and subtract* facility enables addition and subtraction to be performed directly on binary-coded decimal numbers.

PART II

PROGRAMMING

1. PROGRAMMING SYSTEMS AND SUBROUTINES

1.1 General

To assist the user of the computer, a wide and expanding range of programming aids is available. These include several assemblers and compilers as well as a simulator for the I.B.M. 650. In addition, there are a number of programs for general use such as those for the solution of linear programming problems using the symmetric method, multiple regression analysis, and correlation analysis. At a lower level, there is a well developed library of subroutines available to cover more specific requirements in the mathematical analysis field as well as utility programs for general purpose input-output operations, dumping of selected areas of memory, number conversions, and control of peripheral operations.

1.2 Instructions

The computer and peripheral units operate under the programmed control of over one hundred instructions. These instructions are classified into the following categories:

1. Arithmetic
2. Data Transfer
3. Shift
4. Internal Test-and-Branch
5. Console Operation
6. Paper Tape Input-Output
7. Card Input-Output
8. High Speed Printer
9. Magnetic Tape
10. Mass Random Access File
11. Document Sorter

(A partial list of these instructions appears in Appendix II)

Instructions obeyed by the computer must be in binary notation, but the user does not code his programs in this form. Rather, machine running programs are created by use of the General Assembly Program developed by the manufacturer of the computer.

1.3 General Assembly Program

The General Assembly Program (GAP) is a basic assembly program with error checking features and provision for program modification. Input programs are written in a flexible pseudo-code, which is translated by the assembly program into the absolute code of the computer. For example, the

instruction for addition is written by the programmer using the mnemonic "ADD" which is later translated by the assembly program into the proper binary bit configuration for the computer, thus reducing the clerical effort of the programmer. The mnemonic operation codes are chosen for their significance to the user: "ADD" is the mnemonic for the addition instruction, "SUB" is the mnemonic for the subtract instruction, and so forth.

Symbolic notation is also used in assigning memory addresses. The programmer may designate any memory location in decimal notation, and it will be converted to binary by the assembly program. Using alphanumeric notation, the programmer may represent memory addresses by names chosen for his own convenience and the assembly program assigns the memory location. In addition, relative addressing is possible, i.e., if A is the symbolic designation for a memory location, then A + 10 may be used to designate the tenth location from A. Both the one-address format and the general structure of a computer instruction are retained in the instruction portion of the pseudo-code. One instruction written in the pseudo-code is usually translated into one computer instruction.

The user of the computer writes his program using the pseudo-code which is read into memory along with the General Assembly Program itself. The output is a deck of cards or a tape containing the user's original program translated into the absolute code of the computer. This program is then ready to read into memory for execution.

A wide variety of errors - particularly clerical errors and errors due to carelessness on the part of the coder - can be detected during assembly. The removal of the majority of these "trivial" errors, prior to actual debugging, results in a substantial saving in the initial debugging effort.

1.4 WIZ Compiler

The WIZ Compiler is a simplified automatic coding technique using an algebraic type of language. Input, output and computations of sine, cosine, $\log_{10}$, $\log_e$ exponential, arctan, integer conversion, and absolute value of an argument, are provided as built in subroutines which may be called by a WIZ program. The WIZ program is translated by the machine in a single run into the absolute code of the computer at the average rate of 100 cards per minute. GAP coded subroutines may also be incorporated in a WIZ coded program.

```
ROOT  =  (-B + SQRT.(B*B-4.*A*C))/(2.*A)
```

Examples of WIZ language.

1.5 WIZOR Compiler

The WIZOR compiler is similar in language specification to WIZ but allows greater flexibility for dealing with a wider range of input and output information, as well as allowing the user to manipulate individual binary digits of a machine word. It may be used as a compiling compiler in the sense that WIZOR will allow the user who wishes to specify his own compiler language to construct a compiler for his language in WIZOR code, thus greatly simplifying the writing of new compilers. In the following example, the initial symbol 'B' indicates that the statement relates to logical 'and' and 'or' operations on the bits of machine words.

```
B        ((A+B*C)*D+E)*-Z
```

Example of WIZOR coding

1.6 ZOOM Compiler

The ZOOM compiler is essentially a macro instruction generator in that groups of common operations in GAP code may be defined by a suitable name and later recalled for inclusion in the program assembly by use of the chosen name in a ZOOM compilation. In general, ZOOM produces a GAP language output so that subsequent alterations may be made to the final program in GAP language if desired, and a machine language program assembled using GAP.

```
S        X*(A/S = J)-R = L
G        SEN1, LDA L.
         STA RESULT 1.
```

Example of ZOOM coding.

The ZOOM Coding System is of particular use for the construction of transformation programs associated with a program for multiple linear regression computations. This latter program provides cross referencing information in ZOOM coding for transformations in the multiple linear regression program.

1.7 General Compiler*

The General Compiler (GECOM) is an automatic coding technique which uses a flexible source language to describe the problem and the data. The compiler language is completely symbolic, has variable format, and allows GAP language to be interspersed as required in the problem description. English language commands for the General Compiler can be in COBOL, ALGOL, FORTRAN, or GECOM.

COMPUTE INCOME~TAX FROM (GROSS~PAY - (DEPENDENTS * 13.00)) * 0.18.	A(I,J) = SIN (A - B) + (33.33 * (Q) (P-1) #3) - LOG (A - Q).

Examples of GECOM Language.

**It should be noted that the General Compiler requires the use of a minimum of four magnetic tape decks and therefore will NOT be available with the system at present installed by the University.*

In its most general form, the General Compiler language consists of three parts: namely, Procedure Statements, Data Description, Environment. Collectively, these three parts constitute a "source program". The computer will have a programmed executive routine or "compiler" which will translate a source program into a machine language program. This machine language program will be known as the "object program". The procedure portion of a source program is a sequence of statements which specify the steps that must be followed to solve a problem. The statements resemble English sentences in that their wordage is readable and meaningful.

The system will handle all types of library routines. Generators as well as open and closed fixed sub-routines are supplied in the basic library. Library routines, including routines written in the source language, may reference other library routines to any depth.

1.8 IBM 650 Simulator

To aid those who already have programs written for an IBM 650, a program which simulates the operation of the IBM 650 on the GE 225 is available. Hence a user in such a case can continue to produce results from important programs during the re-programming phase for change-over to the GE 225.

1.9 Basic Mathematical Subroutines.

A wide variety of basic mathematical routines is available for the computation of elementary mathematical functions, solution of non-linear equations, and curve fitting. Many are available in either single precision, double precision, or floating point mode. The list includes:-

Sine A

Cosine A

Tangent A

Arc Tangent A

2, 10 and e raised to the power A

Log_2, Log_{10}, and Log_e of A

Square Root of A

pth root of A

Roots of a polynomial of degree N

Solution of simultaneous Equations

Inversion of a matrix.

Least Squares Polynomial curve fitting.

1.10 Utility Routines

To facilitate communication between the computer, which manipulates binary information, and the programmer who works and thinks with more compact symbols (decimal numbers, letters, punctuation marks, etc.), the following utility routines are provided:

1. Decimal integers to binary integers.
2. Decimal fractions to binary fractions.
3. Binary words to octal words (octal numbers are convenient symbols for representing binary numbers).
4. Binary integers to decimal integers.
5. Binary fractions to decimal fractions.
6. Single and Double Precision scaling routines that accept variable sized decimal numbers as input and produce binary numbers (either one or two word) with a specified binary scale as output.

In addition, several complete input/output subroutines are available, and these subroutines cover in one package all standard input/output requirements.

Other utility routines include:

1. Standard routines to load instructions into memory from cards or tape.
2. Standard routines for use in program debugging to punch out memory (or selected areas of memory) on cards, to print out using the High Speed Printer, or to write memory on tape.
3. A standard set of punched paper tape data input and output routines.
4. A standard set of punched card data input and output routines.
5. A standard set of magnetic tape data input and output routines.

PART III

BUILDING AND SERVICES

1. GENERAL

The building was specially designed as a lower ground floor to the Faculty Office of the Main Engineering Building, St. Lucia. The net area is 3,000 sq. feet.

The north and east walls are retaining walls, natural light being available from the south. In planning the layout, particular attention was paid to the service function of the Centre which makes it preferable to limit direct access to the computer room and programmers room, and to a less extent to the data preparation room. A feature of the building is the clients' room which enables three groups of users to check and discuss their work while being able to observe the computer in operation.

Normal provision is made for Computer Centre staff, including a tea room and the usual amenities.

A plant room houses the air conditioning plant, motor alternator for 50 c.p.s./60 c.p.s. conversion and electrical switchboards.

Access to the building can be obtained from road level (Circular Drive) through the Faculty Office block or at lower ground floor level.

The floor of the computer room comprises multiple removable sections, each of which can be raised independently. Conditioned air is supplied from beneath each unit using suitable divided spaces between the floor and the concrete sub-floor as ducts. All wiring to and from computer units is also arranged below floor level.

Acoustic treatment of the ceiling of all rooms is provided and a false ceiling has enabled all lighting fittings to be recessed and overhead air conditioning ducts concealed.

2. AIR CONDITIONING PLANT

Within the Computer Centre there are three distinct zones served by the air conditioning plant, as follows:

(1) The Computer Area - comprising Computer Room, Maintenance and Data Preparation and Tape Store. Conditions within these rooms are maintained at 75°F d.b.t. and 45% R.H.

(2) Computer Equipment - The supply air to the equipment is maintained at 65°F d.b.t. and 63% R.H., i.e. the moisture content of the air supplied to the equipment is equal to that of the air within the rooms at the conditions in (1) above.

(3) Associated Offices - comprising Lecture Room, Clients, Lobby, Staff Room, O.I.C., Engineer, Programmers and Library. Comfort conditions are maintained in these rooms viz. 75°F d.b.t. and 55% R.H. Three sub zones occur within this major zone.

The tolerance on the above temperatures and relative humidities is ±5%.

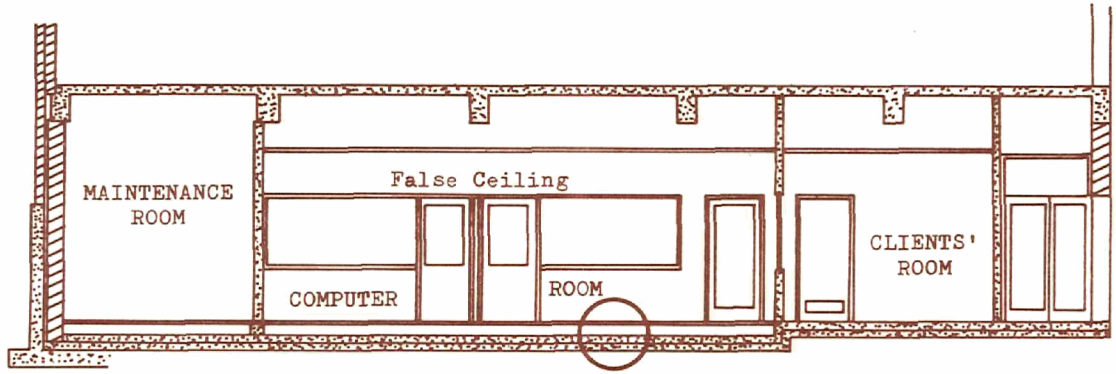
Each system is arranged as an unattended plant, operated from the computer room. Plant A supplies zones 1 and 2 and Plant B supplies zone 3, the capacities being 148,000 BTU/hr. and 112,000 BTU/hr. respectively.

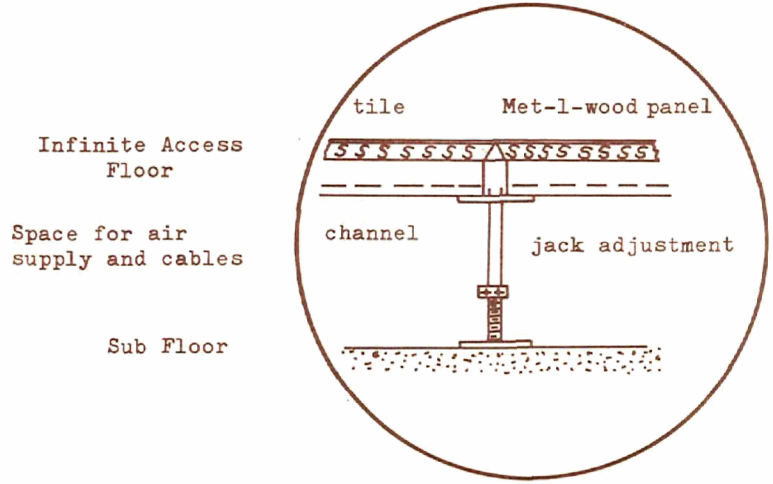
Reliability of operation of air conditioning plant A, supplying the computer units, is of the utmost importance. Should failure of the refrigerating system of this plant occur, a changeover to Plant B compressor can be effected immediately. Under emergency conditions, flow of air, irrespective of its condition, enables the operation of the computer to continue and hence arrangements can be made to run Plant A as a ventilating system only.

3. ELECTRICITY SUPPLY

Normal 415/240 volt 50 cps. supply is provided from an adjacent substation, particular care being taken to avoid interruption. Power for the computer itself is supplied from a 30 KVA motor generator set. This set, which provides a 3 phase, 120/208 volt, 60 cycles/sec supply, is necessary to ensure constant voltage free from line transients which could affect the operation of the computer. The capacity of the motor generator set is considered to be sufficient to cater for any likely future expansion of the computer facilities. The total connected load to the Computer Centre is:-

	KVA
Lighting	19.6
Air Conditioning Plant	55.0 (est.)
Computer Supply	14.5
Data Preparation Equipment and Miscellaneous	10.4
Total	99.5
Average load	46.5 KW



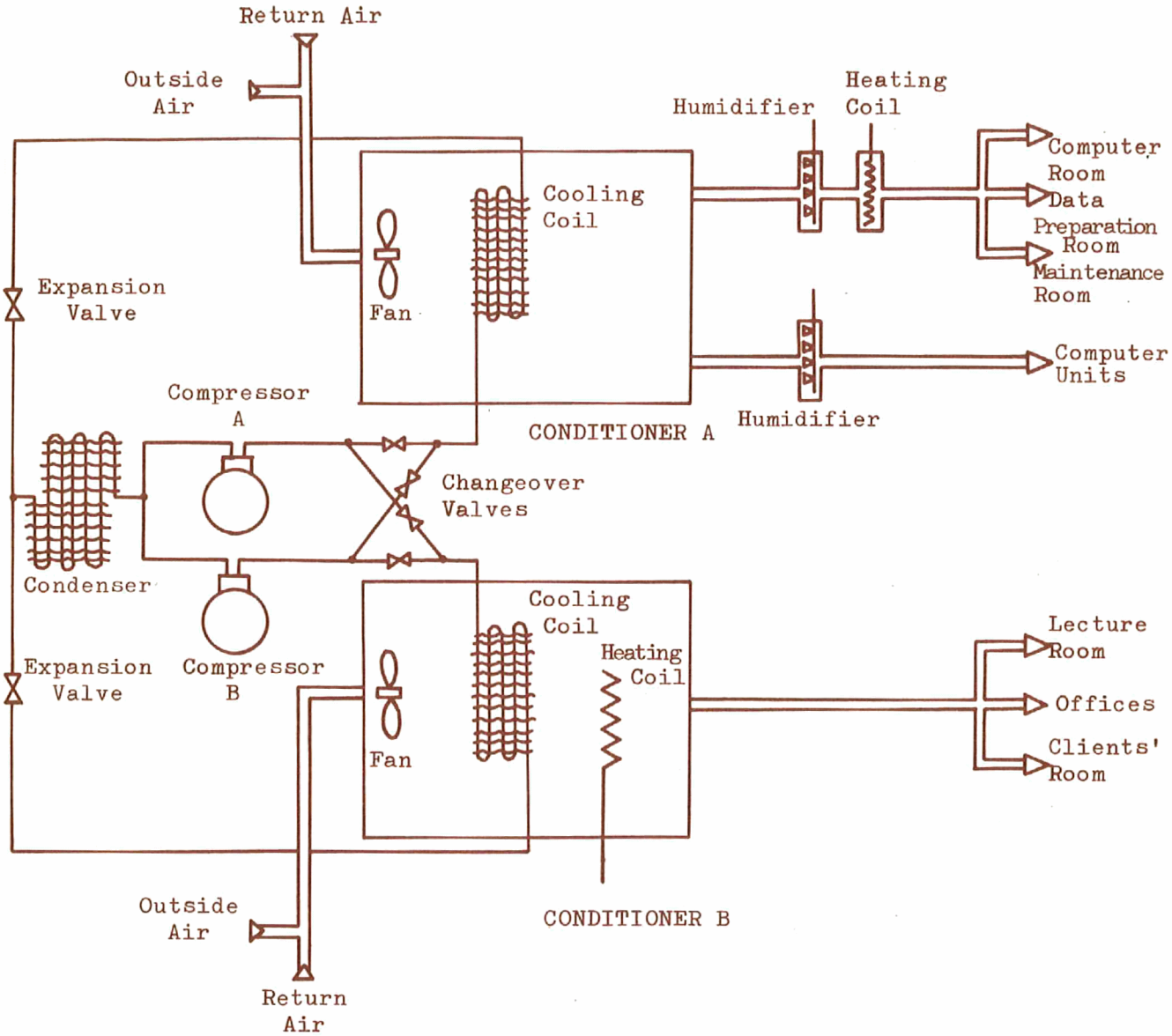


FLOOR DETAIL

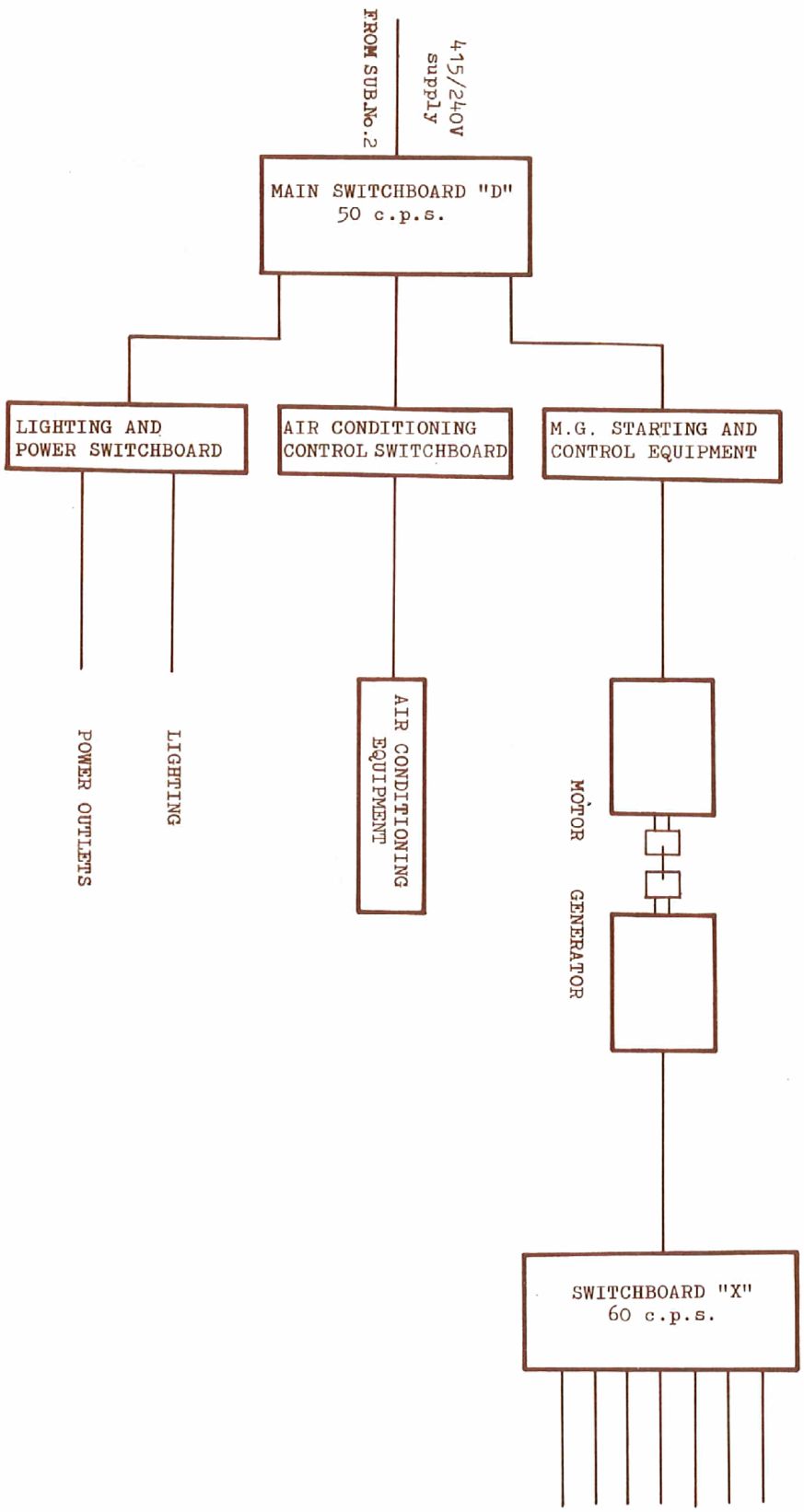
8 6 4 2 0 8 16 32

Scale of Feet

COMPUTER CENTRE
SECTIONAL ELEVATION



SIMPLIFIED SCHEMATIC DIAGRAM OF AIR CONDITIONING EQUIPMENT



SCHEMATIC DIAGRAM OF 50 c.p.p. and 60 c.p.s. POWER SUPPLIES - COMPUTER CENTRE

APPENDIX I - CHARACTER REPRESENTATION

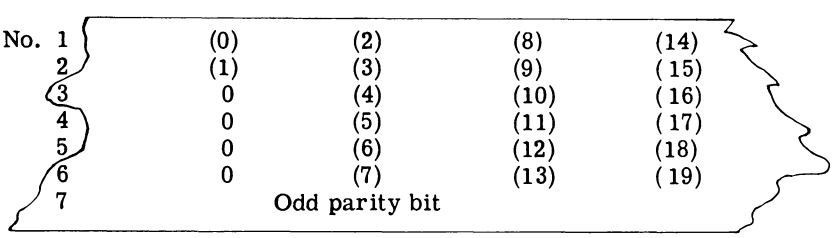
Character	Hollerith Code (Punch in Rows:)	BCD Memory (Octal)	BCD Tape (Octal)
0	0	00	12
1	1	01	01
2	2	02	02
3	3	03	03
4	4	04	04
5	5	05	05
6	6	06	06
7	7	07	07
8	8	10	10
9	9	11	11
Unassigned		12	
#		13	13
@		14	14
Unassigned		15	15
=	6-8	16	16
Unassigned		17	17
+	12	20	60
A	12-1	21	61
B	12-2	22	62
C	12-3	23	63
D	12-4	24	64
E	12-5	25	65
F	12-6	26	66
G	12-7	27	67
H	12-8	30	70
I	12-9	31	71
Unassigned		32	72
.	12-3-8	33	73
□	12-4-8	34	74
Unassigned		35	75
Unassigned		36	76
Unassigned		37	77
—	11	40	40

Character	Hollerith Code (Punch in Rows:)	BCD Memory (Octal)	BCD Tape (Octal)
J	11-1	41	41
K	11-2	42	42
L	11-3	43	43
M	11-4	44	44
N	11-5	45	45
O	11-6	46	46
P	11-7	47	47
Q	11-8	50	50
R	11-9	51	51
Unassigned		52	52
$	11-3-8	53	53
*	11-4-8	54	54
Unassigned		55	55
Unassigned		56	56
Unassigned		57	57
None (Space)	Blank Column	60	20
/	0-1	61	21
S	0-2	62	22
T	0-3	63	23
U	0-4	64	24
V	0-5	65	25
W	0-6	66	26
X	0-7	67	27
Y	0-8	70	30
Z	0-9	71	31
Unassigned		72	32
,	0-3-8	73	33
%	0-4-8	74	34
⊏	0-5-8	75	35
⊐	0-6-8	76	36
Unassigned		77	37

The magnetic tape has 7 tracks for recording. Six bits of Binary or BCD data are recorded simultaneously in 6 tracks laterally across the magnetic tape to to form a tape character. The seventh track contains a parity bit for each tape character, generated at time of recording. In Binary mode of operation, odd parity is used. That is, the number of ones in tracks 1-7 will always be odd. In BCD mode, even parity is used.

In the binary coded decimal mode of operation each binary coded decimal character is stored on tape as a corresponding magnetic tape character; that is, a memory word is stored as three magnetic tape characters.

In Binary mode of operation, the 20 bits of a memory word are written on magnetic tape as 4 magnetic tape characters. Three of the magnetic tape characters contain 6 bits of data each, while the fourth magnetic tape character contains only 2 bits of data. The four remaining bits in this character will be written as zeros. These 4 zeros will automatically be inserted when recorded and ignored when read back from tape. In the figure below, the numbers in parenthesis are the bit portions of the word.



APPENDIX II - INSTRUCTION REPERTOIRE

The following is a list of the GE 225 instructions and a brief description of their function:

LEGEND

rA = A register or accumulator
rQ = Q register for double length operations
Y = Memory location
rN = Output register for typing
Word X = Automatic modification of address word
( ) = Contents of

Mnemonic	Description	Execution Times (microseconds)
	ARITHMETIC COMMANDS	
ADD	Add (Y) to (rA). Result is in rA	36
SUB	Subtract (Y) from (rA). Result is in rA	54
MPY	Multiply (Y) by (rQ). Result is in rA and rQ	414 (max.)
DVD	Divide (rA) and (rQ) by (Y). Quotient is in rA, remainder is rQ.	522 (max.)
DAD	Double Length Add (Y) and (Y + 1) to (rA) and r(Q)	54
DSU	Double Length Subtract (Y) and (Y + 1) from (rA) and (rQ)	90
FAD	Floating Point Add	700 (max.)
FSU	Floating Point Subtract	700 (max.)
FMP	Floating Point Multiply	1053 (max.)
FDV	Floating Point Divide	1205 (max.)
	INFORMATION TRANSFER COMMANDS	
LDA	Load rA from Y	36
STA	Store (rA) in Y	36
DLD	Double Length Load - load rA and rQ from Y and Y + 1	54

DST	Double Length Store - rA and rQ are stored in Y and Y + 1	54
FLD	Floating Point Load	72
FST	Floating Point Store	72
MAQ	Move (rA) to rQ	54
LQA	Load rQ from rA	54
XAQ	Exchange (rA) and (rQ)	54
LAQ	Load A from Q	54

SHIFT COMMANDS

SRA	Shift (rA) right (open-end)	36 +
SCA	Shift (rA) right (circular)	36 +
SRD	Shift right double -(rA) and (rQ) shifted right (open-end)	36 +
SCD	Shift right circular double - (rA) and (rQ) shifted right (circular)	36 +
SAN	Shift (rA) and (rN) right - (open-end)	36 +
SNA	Shift (rN) and (rA) right - (open-end)	36 +
NAQ	Shift (rN), (rA) and (rQ) right - (open-end)	36 +
ANQ	Shift (rA) into (rN) and (rQ)	36 +
SLA	Shift (rA) left (open-end)	36 +
SLD	Shift left double - (rA) and (rQ) shifted left	36 +
NOR	Normalize (A) Register	54 +
DNO	Double Length Normalize - (rA) and (rQ)	54 +

LOGICAL AND MODIFY COMMANDS

STO	Store Operand Address from rA in Y	54
ORY	OR (rA) into Y	54
EXT	Extract into rA	54
INX	Increment (Word X) by a constant contained in the INX command	54
LDZ	Load Zeros into rA	54
LDO	Load a One into rA	54
LMO	Load Minus One into rA	54
ADO	Add One to (rA)	54

SBO	Subtract One from (rA)	54
NOP	No Operation - Zero added to (rA)	54
CPL	Complement (rA) - invert bits	54
NEG	Negate (rA) - subtract (rA) from 0	54
CHS	Change Sign of rA	36

TEST AND BRANCH COMMANDS

BRU	Branch Unconditionally	18
SPB	Store Location and Branch Location of this command stored in Word X	36
BXH	Branch if (Word X) High (greater than or equal to a constant contained in the BXH command)	54
BXL	Branch if (Word X) Low (less than a constant contained in the BXL command)	54
BZE	Branch on Zero (contents of rA is zero)	36
BNZ	Branch on no Zero (contents of rA is not zero)	36
BPL	Branch on Plus (sign of rA is +)	36
BMI	Branch on Minus (sign of rA is -)	36
BOV	Branch on Overflow (if overflow indicator is on)	36
BNO	Branch on no Overflow (if overflow indicator is off)	36
BOD	Branch on Odd (contents of rA is odd)	36
BEV	Branch on Even (contents of rA is even)	36
BPE	Branch on Parity Error	36
BPC	Branch on Parity Correct	36

EXTERNAL EFFECT AND INPUT-OUT COMMANDS

OFF	Power supplies for typewriter, paper tape reader and punch are turned off	36
TON	Turn on power for typewriter	36
RCS	Read Control Switches - a word will be stored in rA corresponding to control switches	36
TYP	Type contents of rN	36

PON	Turn power on for Paper Tape Punch	36
WPT	Write Paper Tape - punch contents of rN	36
RCB	Read Card Binary - continuous read	36
RCD	Read Card Decimal - continuous read	36
HCR	Halt Card Reader	36
WCB	Write Card Binary	36
WCD	Write Card Decimal	36
RON	Turn on power for Paper Tape Reader	36
RPT	Read Paper Tape into rN	36
BCR	Branch if Card Reader is ready	36
BCN	Branch if Card Reader is not ready	36
BPR	Branch on Card Punch ready	36
BPN	Branch on Card Punch not ready	36
BNR	Branch on rN ready	36
BNN	Branch on rN not ready	36
SEL	Select Controller number N (the next machine words will be sent to the controller nominated)	36
SLW	Slew paper N lines	36
SLT	Slew to channel K of the control tape	36
WPL	Write a print line starting from location Y	36
WFL	Write a format line, formal control starting at location Y (must be followed by a WPL)	36

Note that these instructions generate two instructions in memory and must be preceded by an SEL.

BCS	BZE	Branch on Printer Error	36
BCS	BNE	Branch on printer error	36
BCS	BNP	Branch on printer not out of paper	36
BCS	BOP	Branch on printer out of paper	36
BCS	BPN	Branch on printer not ready	36
BCS	BPR	Branch on printer ready	36
		Auxiliary Arithmetic Unit Branch and Setting Commands	
BAR	BAN	Branch on AAU not ready	36

BAR	BAR	Branch on AAU ready	36
BAR	BER	Branch on AAU error	36
BAR	BMI	Branch on AAU Minus	36
BAR	BNE	Branch on AAU no error	36
BAR	BNO	Branch on AAU no overflow	36
BAR	BNU	Branch on AAU no underflow	36
BAR	BNZ	Branch on AAU not zero	36
BAR	BOV	Branch on AAU overflow	36
BAR	BPL	Branch on AAU plus	36
BAR	BUF	Branch on AAU underflow	36
BAR	BZE	Branch on AAU zero	36
SET	NFLPOINT	Set to normalized floating point mode	36
SET	UFLPOINT	Set to unnormalized floating point mode	36
SET	FIXPOINT	Set to fixed point mode	36

Note that BCS and BAR commands require a controller priority number in the index register field.

Commands for other peripheral equipment such as magnetic tape and mass random access file memory are not included here.