1956 First STRETCH Proposal

PRELIMINARY DESCRIPTION

OF PROPOSED

MULTIPLEX 10 MEGAPULSE AUTOMATIC COMPUTER

This document contains information of a proprietary nature. Since this information can affect the competitive position of IBM, it is requested that the contents be kept in confidence.

International Business Machines Corporation February 27, 1956

TABLE OF CONTENTS

Summary Specifications

- 1.1 Radix
- 1.2 Word Size
- 1.3 Primary Arithmetic System
- 1.4 Secondary Arithmetic System
- 1.5 Editing System
- 1.6 Selector System
- 1.7 Logical Control System
- 1.8 Magnetic Core Memory
- 1.9 Ultra-Fast Memory
- 1.10 Magnetic Disc Memory
- 1.11 Input-Output Computer Section
- 1.12 Type 727 Magnetic Tape Units
- 1.13 Faster Magnetic Tape Units
- 1.14 Electronic Printer Plotter
- 1.15 Mechanical Printer
- 1.16 Card Reader
- 1.17 Card Punch
- 1.18 Manual Keyboard and Typewriter
- 1.19 Programming
- 1.20 Auto Checking and Error Correction
- 1.21 Unattended Operation
- 1.22 Fault Location
- 1.23 Solid-State Components
- 1.24 Table of Characteristics

Organization of Information

- 2.1 Word Size
- 2.2 Data Words
- 2.3 Instruction Words
- 2.4 Logical Words

Memory

- 3.1 Main Memory
- 3.2 Fast Memory
- 3.3 Ultra-Fast Memory
- 3.4 Memory Addressing

Arithmetic and Control System

- 4.1 Primary Arithmetic System
- 4.2 Addition and Subtraction
- 4.3 Multiplication
- 4.4 Division
- 4.5 Multiple Precision
- 4.6 Shifting
- 4.7 Control Decoder
- 4.8 Control Accumulator
- 4.9 Program Counter

Instruction System

- 5.1 Arithmetic Command Set
- 5.2 Register to Register Transfers
- 5.3 Shift Operations
- 5.4 Control Operations

Selectors

*

- 6.1 Selectors
- 6.2 Selector Instructions
- 6.3 Logical Connectives
- 6.4 Real Time Input-Output
- 6.5 Example of Use

Editing

7.1 Editing Example

Input-Output

- 8.1 Modes of Operation
- 8.2 Input-Output Interchanger
- 8.3 Input-Output Computer
- 8.4 Input-Output Addressing
- 8.5 Input-Output Multiplexing
- 8.6 Input-Output Console
- 8.7 Tape Units
- 8.8 Magnetic Disc Memory
- 8.9 Electronic Printer-Plotter
- 8.10 Mechanical Printer
- 8.11 Card Reader
- 8.12 Card Punch
- 8.13 Interrogation Typewriter

Error Detection and Correction

- 9.1 Automatic Checking
- 9.2 Automatic Error Correction

Engineering

•,

î

- 10.1 Automation of Design
- 10.2 Reliability
- 10.3 Circuit Design
- 10.4 Modular Design

Maintenance

- 11.1 Trouble Prediction and Location
- 11.2 Marginal Checking
- 11.3 Monitor Circuits
- 11.4 Error Location Circuits
- 11.5 Trouble Correction

SUMMARY SPECIFICATIONS

The new scientific computer described in these specifications is intended to handle mathematical and logical problems with efficiency and flexibility. It employs a parallel arithmetic system and multiple high speed magnetic core memories, and it is capable of handling a variety of input-output devices, as well as communication over telephone lines with input-output units at remote points. Auxiliary equipment is available for the handling of supplementary operations, such as card to tape and tape to printer conversions.

The machine will be designed to operate for extended periods unattended by either operating or engineering personnel. The reliability necessary for this type of operation will be achieved by the use of solid state components, by the inclusion of automatic error correction in important areas, and by a fault prediction system which will be an integral part of the design. The automatic control features which contribute to the ability to operate unattended include automatic error detection with provision for control of the program by the error detecting circuits, automatic means by which the machine can assign input-output units to first one task and then another as required, automatic means for switching from one problem to another as they are completed or reach a point requiring operator attention, and automatic means for operating on more than one problem at the same time.

A new concept of machine control permits the several memory units, the input-output units, and the arithmetic systems of the computer to operate concurrently. Secondary arithmetic systems are provided for the control of input-output and for the modification of instruction addresses, allowing the primary arithmetic system to process useful data without interruption.

The design of this machine embodies many new principles, techniques and components. It is IBM's intention to state its design objectives in terms of delivery date and level of performance which can be realized by a bold, progressive, yet scientifically sound program. It is expected that continuing research during the development period of the machine will result in substantial improvements over the objectives stated here. These may well result in the availability of even higher arithmetic speeds, larger or faster memories, faster internal controls in the computer, or new forms of input-output.

)))

1.1 Radix

The choice between a decimal and a binary radix for the arithmetic system and for memory addressing is to depend upon the results of application analysis by an IBM study group. Their decision is to be subject to the concurrence of the Los Alamos Scientific Laboratory.

This proposal has been written as though the binary approach were to be used. If the machine should prove to be decimal, the capacities of the memories and related mechanisms would be modified to the nearest decimal equivalent.

It should be understood that the machine is designed to operate efficiently upon data with any radix, and that it is expected to be highly effective in handling both binary and decimal data.

Ì

1.2 Word Size

7

A word will consist of sixty binary information bits. To this will be added additional bits for checking and error correction. The manner in which the bits are allocated for use as data and instruction words will depend in part upon whether the machine is binary or decimal. If binary, data words will consist of a forty-eight bit fraction, a ten-bit exponent, a one-bit exponent overflow tag, and a one-bit sign. The format of instruction words will vary for different classes of instructions, and will be governed by the information requirements of each class.

1.3 Primary Arithmetic System

The operating speeds of the primary arithmetic system are as follows:

Addition and Subtraction	0.2 microsecond fixed point 0.6 microsecond floating point
Multiplication	<pre>1.2 microseconds (nominal, fixed or floating point)</pre>
Division	1.8 microseconds

(nominal, fixed or floating point)

1.4 Secondary Arithmetic System

A secondary arithmetic system will be provided in association with the instruction decoder to perform the address arithmetic required for indexing and similar control functions. Address arithmetic will be performed concurrently with the operation of the primary arithmetic unit.

1.5 Editing System

An editing system will be provided to arrange input data for use by the computer and to convert computer results into the form required by output devices. It will be capable of converting data from one radix to another as it enters and leaves the computer for those cases in which the input-output radix and computer radix are not the same. It will similarly translate data from any one machine code to any other as it flows in and out of the machine, permitting direct input of data from instruments and from other machine systems which do not use the same code language as the computer. It will also compress input data and instructions into the more compact format of the data words used within the machine.

1.6 Selector System

The computer will provide programmed access to a large number of selector devices for use in logical control. Some of these selectors will be set by the status of parts of the machine; some will be set by the data in the several machine registers; and other groups of selectors can be set from the operator's console or by the program. Selectors will be provided which can be controlled from, and will provide outputs to remote locations.

-4

1.7 Logical Control System

The computer will include new and powerful logical instructions. A special logical control system will permit any of the sixteen logical connectives to be executed under program control upon the sixty bits of each of two data words to produce a third sixty bit word representing the logical result.

A new form of logical instruction will permit the examination of statements of logical algebra in alternational normal form.

1.8 Magnetic Core Memory

The addressing system will provide for direct addressing of up to 1,048,576 words of memory. The memory is of three types. The quantity of memory of each type can be arranged to suit the requirements of the user. The largest of the memory types will have a full read-write cycle time of 2.0 microseconds and will be assembled in units of 8,192 words. A smaller and faster memory will have a full read – write cycle time of 0.5 microsecond and will be assembled in units of 0.5 microsecond and will be assembled in units of 512 words. In addition,

The individual memory units will be capable of operating concurrently and asynchronously with the result that the average rate at which memory references can take place will be much higher than the rate for any one memory.

1.9 Ultra-Fast Memory

The magnetic core memories will be supplemented by at least sixteen words of 0.2 microsecond memory for data which is used with great frequency by the program.

1.10 Magnetic Disc Memory

Provisions will be made for magnetic disc memories to supplement the magnetic core memory for the storage of data and instructions for large problems. Magnetic disc memory will be available in blocks of 1,048,576 words. Information will flow between the disc memory and the computer at the rate of 15,000,000 information bits a second, or one word each 4 microseconds. The arrangement of the magnetic heads will be such as to permit continuous flow of information as the magnetic reading and writing proceeds from one track to another on the disc memory.

1.11 Input-Output Computer Section

1

The input-output computer will function as an integrated communications system between the input-output devices and the several memories. It will be provided with its own arithmetic unit and control mechanism, so that address computations, editing, radix conversion, scaling, code conversion and similar operations relating to input and output will not interrupt primary computation. In addition, it will control communication between the input-output units and memory over the bus system and between one input-output unit and another. The following paragraphs provide a partial list of the input and output units which can be used with the computer. In addition to these and other devices, the computer will be capable of communicating over wire lines with instruments and control devices at remote points.

1.12 Type 727 Magnetic Tape Units

Provision will be made for use with the computer of the present IBM Type 727 Tape Units. These tapes will provide the communication link between the computer and Type 704 and Type 705 EDPM machines and their auxiliary equipment.

1.13 Faster Magnetic Tape Units

Provision will be made for equipping the computer with new magnetic tapes. These tapes will have an information flow rate not less than ten times that of the present Type 727 Tape Units. The exact specifications for these tapes require further study before being agreed upon.

1.14 Electronic Printer – Plotter

The computer will handle an electronic printer capable of recording 16,500 characters a second. This printer will operate both as a page printer and as a point plotter. It will record on microfilm. The microfilm image can subsequently be transcribed to conventional paper forms through a separate copying device which can produce one or as many copies as are desired. A secondary visual display can be provided for operator surveillance of the operation of the machine.

1.15 Mechanical Printer

Provision will be made for the IBM 1000 lines-a-minute wire printer for direct communication with the computer and for use as an auxiliary tape operated printer. It may later be desired to substitute for this printer a magnetic printer with a similar operating speed.

1.16 Card Reader

Provision will be made for a 500 cards-a-minute parallel card reader for direct attachment to the computer, or for use as an auxiliary card to tape converter. The speed of the card reader may later be revised upward to 1000 cards a minute. 1.17 Card Punch

Provision will be made for a 155 cards-a-minute card punch for direct attachment to the computer, or for use as an auxiliary tape operated punch.

1.18 Manual Keyboard and Typewriter

A manual input keyboard and output typewriter will be provided as a part of the operator's console to facilitate operator control of the machine.

1.19 Programming

New instructions and a new instruction format are required to insure that the computer has the greatest possible effectiveness on large and complex problems. A joint study group, composed of IBM and Los Alamos personnel, would study the several possible forms of instruction and determine which is the most effective for advanced programming.

Among the new features to be controlled for which no precedent exists are the following:

- a. The editing system
- b. The logical control system
- c. The selector system
- d. The indexing system
- e. The indirect address system
- f. The input-output control

Methods of computer assisted programming similar to FORTRAN WILL be developed concurrently with the development of the instruction system, with the participation of the Los Alamos Scientific Laboratory. In this way, it will be assured that the computer assisted programming system and the instruction set are in harmony with one another, and represent the true desires of users. In view of the high speed of the computer, it will be essential that most programs be produced with the assistance of the computer. However, the system employed will be one which will permit the user to readily write and understand the programs used by the machine.

The instructions in the latter part of this proposal are to serve as an example, and do not represent a recommended set.

1.20 Automatic Checking and Error Correction

Automatic checking will be provided throughout the computer. The nature of the check used in each area of the machine will depend upon the requirements of the mechanism. Generally speaking, the check will detect single errors and most multiple errors.

Parts of the machine will be equipped to correct as well as detect errors. In these areas, the mechanism will be designed to detect all single and double errors and most other multiple errors, and will correct single errors.

When an error is detected and corrected, its location and nature will be recorded to assist in the removal of the faulty part at a later time convenient to the user and the engineer.

During addition, subtraction, multiplication and division, the primary arithmetic unit will repeat its operation if an error is detected. If the fault is intermittent and does not occur the second time, the arithmetic unit will produce a correct result and continue on with the problem without further interruption.

1.21 Unattended Operation

It is to be expected that it will be practical to operate the computer for extended periods, such as for example, an entire night shift, unattended by either operating or maintenance personnel. The use of automatic error and checking devices will provide a reasonable guarantee against the occurrence of undetected errors, and will provide centrols for programmed intervention in the event of certain classes of errors. Automatic single error correction in parts of the computer will allow it to continue operation despite faulty parts in these areas and permit the correction of many classes of faults to be deferred to the next scheduled maintenance period. The computer will be capable of executing more than one program at the same time. In the event that a program is completed or requires human intervention, the computer is capable of setting it aside and continuing on with other work. The computer control of the input-output will allow it to select the units to be used at any time, and to reassign the use of the individual units as its requirements change.

Fault Location

1.22

The automatic checking, error correction and marginal checking mechanism within the machine will be specifically designed to reveal the nature and location of machine faults, as well as their existence. This will be done in such a way as to direct the engineer to the source of trouble and permit its correction in a minimum time. The design of the components will be modular. A concerted effort will be made to use the least possible number of different kinds of modules and to use the same module in many places. In this way, most repairs will be made by plugging in one of a small number of replacement modular units.

1.23 Solid-State Components

The high operating speed of the machine has been made possible by the development of a new form of magnetic core and by the high speed, low power consumption, and small size of gaseous diffusion junction transistors. Magnetic cores will provide major memory functions. Transistors supplemented by semiconductor diodes will be used for amplification and switching. The operating frequency of the transistors will depend upon the area of the machine in which they are used. Most will operate at the rate of ten million pulses per second.

1.24 Table of Characteristics

Because of the advance nature of the proposed computer, the details of its specifications cannot be completely settled until the end of the first year of the development program. It is anticipated that further detailing of the machine together with the application analysis will confirm the following list of characteristics:

Radix	Binary or decimal with direct and simple provision for the efficient handling of numbers in any radix
Notation	Fixed and Floating point (normalized and unnormalized)
Word Size	60 binary information bits
Checking	Automatic checking throughout; Automatic single error correction in many areas
Addressing	Memory addressed as though it were a single unit; Capacity will be provided for directly addressing 1,048,576 words. New instructions will facilitate indirect addressing of much larger amounts of memory.
Base Frequency	10 megapulse
Add and Subtract	0.2 microsecond fixed point 0.6 microsecond floating point
Multiply	1.2 microseconds nominal, fixed or floating point
Divide	1.8 microseconds nominal, fixed or floating point
Large Core Memory	Units of 8, 192 words, with a full read write cycle time of 2 microseconds
Fast Core Memory	Units of 512 words, with a full cycle read write time of 0.5 microsecond
Ultra–Fast Memory	Sixteen or more single word registers with an access time of 0.2 microsecond
Magnetic Disc Memory	Units of 1,048,576 word capacity; Continuous 15,000,000 bits per second information transfer for each unit

Type 727 Tape Units	s 1/2" x 2400' magnetic oxide tape; Six information tracks in parallel; A seventh track for checking Speed: 75" per second Density: 200 bits per inch
Faster Magnetic Tap Unit	be This unit will provide 10 or more times the information flow rate of the Type 727 tapes. Details of specifications to be determined later.
Electronic Printer	Usable both as a page printer and as a point plotter 16.5 KC character rate 64 characters 1024 x 1024 raster Automatic checking Records on microfilm Optional visual display
Mechanical Printer	1000 lines a minute 120 characters per line Automatic checking It may later be proposed to substitute a magnetic printer of similar or higher speed.
Card Reader	500 cards per minute Parallel reading Automatic checking A 1000 card per minute Reader may be proposed later.
Card Punch	155 or more cards per minute. Automatic checking.
Console Typewriter	Manual input 10 characters per second output Automatic checking
Auxiliary Equipmen	t Card Readers, Printers, and Card Punches can be obtained for operation with Type 727 Tape Units as auxiliary equipment.

s - +*

1

ORGANIZATION OF INFORMATION

The organization of information in the computer will depend in part on the decision as to whether the arithmetic and memory addressing systems will be binary or decimal. In either case, the design will be such as to permit a high degree of flexibility in the handling of numerical data in any radix, as well as alphabetic data and complex logical expressions. The organization of information will be discussed as though the machine were to be binary.

2.1 Word Size

÷

The sixty information bits in a data word can be employed on a strictly binary basis. When employed in ways which do not involve arithmetic upon the word, the bits can be divided into groups of 3, 4, 5 or 6 bits to represent octal, decimal, alphabetical or alphanumerical characters. The interpretation of the sixty bits in a word will depend upon the use to which the word is put within the machine. The editing system will compress input data and instructions and expand results for read out from the machine.

2.2 Data Words

Numerical data words will ordinarily be expressed as a forty-eight bit fraction, a ten bit exponent, a single bit sign for the fraction and a single bit exponent overflow tag. The overflow tag will indicate whether the exponent of the number has at any time exceeded the bounds of the ten bit field allowed for it. Alphabetic data can be recorded by dividing the sixty bits into ten characters of six bits each.

2.3 Instruction Words

The instructions in each of several classes will differ in format in order to include in each a maximum amount of information. The original recording and printing of instructions will ordinarily be in octal or decimal notation. Since many short instructions are required for multiple level indexing and similar control functions, it is probable that some instructions will be one-half word in length. If so, the necessary mechanism will be provided for addressing and manipulating half words.

2.4 Logical Words

Any data or instruction word can be manipulated as a sixty bit logical expression. In addition, the machine is able to extract the settings of its various signal and control devices (selectors) as sixty bit binary words of data for programming as logical expressions. Each word of this type will represent sixty selectors within the machine, with the 0 or 1 condition of each bit indicating the on or off condition of the device.

3.1 Main Memory

1

1

The main memory of the machine will be made up of magnetic cores. Each memory unit will contain 8, 192 words of information. Associated with each memory is a data register which holds the word being stored or read. An additional address register retains the address of the memory reference being made.

The full cycle time of the memory is 2.0 microseconds. Data being read from memory will be available for use at the end of the first 0.8 microsecond of the memory cycle. Data being stored in the memory need not be available in the memory register at the beginning of the memory cycle, but must have been transferred to it by the end of the first 0.8 microsecond. Data words are transferred between the memory register and the other registers of the machine in 0.2 microsecond.

Each of the memory units will operate concurrently and asynchronously with the other memories and with other parts of the computer. Their operation is governed by the primary and by the input-output systems, both of which regulate the flow of information between the memory registers and the remainder of the machine over the input-output and computation bus systems.

Memory will include additional bits for checking and possibly autocorrection. The memory address and the address lines actuated in the memory are checked for errors, also. An automatic marginal checking system within each memory will signal if an element in the driving mechanism does not satisfy the proper standards for operation of the memory. Except in the case of an abrupt failure, this signal is provided sufficiently in advance of the time at which the part can cause a memory failure to permit the replacement of the part in the next routine servicing period. All of the above checks are accomplished automatically.

3.2 Fast Memory

The main memory of the machine is supplemented by a faster form of magnetic core memory. Each of these faster memories contains 512 words.

The full cycle time of the fast memories is 0.5 microsecond. When reading, the data is available for use in its data register at the end of 0.2 microsecond. When writing, the information to be recorded need not be in the data register until 0.2 microsecond after the writing cycle has been initiated.

As in the case of the main memories, each fast memory is provided with data and address registers. The fast memories are capable of operating concurrently with one another and with the main memories.

-3.1-

The fast memories can be used to store instructions and data which are frequently required by the computer, and in this way play a vital part in obtaining a high level of performance. The provisions for checking in the fast memories are the same as those in the main memories.

3.3 Ultra-Fast Memory

;

The main and fast memories will be supplemented by at least sixteen single word registers with a full cycle time of 0.2 microsecond. These registers are needed for the storage of data used frequently by the arithmetic and control mechanisms.

3.4 Memory Addressing

The addressing system allows for the ultimate use of 1,048,576 words of memory. This memory can be divided as needed among large memory units, fast memory units and the ultra-fast memory. In order to automatically distribute the memory references among the several large memories and correspondingly among the fast memories, one or more of the lowest order binary digits in the memory address will identify a memory unit. If two memories of the same type are attached to the machine, the lowest order bit will select between one and the other. so that consecutive addresses will refer to alternate units. If four memories of the same type are employed, the two lowest digits will define the memory unit, so that any four consecutive addresses will be distributed among the four memories. This system has the virtue that it distributes memory references equitably among the several memory units without planning on the part of the programmer. It can only be used to the extent that the number of memories of the same type is a power of two. If, for example, five large memories are attached to the machine, the first four will have their addresses interleaved in the above manner, and the fifth will contain consecutively addressed words.

The addresses used for memory can be a consecutive series of numbers beginning with the ultra-fast memory at the lower extreme and proceeding through the fast memories to the large memories at the higher end of the series. The advantage in the use of consecutive addresses is that programs can be written in which a section falls partly in one memory and partly in another. Additional memory of each type can be attached to the machine at any time. If this is done, the instruction addresses in previously written programs may have to be reassigned through a computer process.

ARITHMETIC AND CONTROL SYSTEM

A new concept of computer control will be employed. The computer will be divided into a number of sections, each of which is capable of operating concurrently with the others. Two secondary arithmetic and control systems, one for the input-output and one for the problem proper, will govern the order in which each part of the machine executes the succession of tasks required of it. The major parts of the machine which are able to operate in this way include:

- 1. The large memory units
- 2. The fast memory units
- 3. The ultra-fast memory
- 4. All of the input-output units, including magnetic tapes, magnetic discs, card reader, card punch, printer and interrogation typewriter
- 5. The primary arithmetic system
- 6. The control decoder
- 7. The input-output computer section
- 8. The input-output address accumulator

The control mechanism assigns tasks to the individual sections of the machine in the order in which they must be executed to maintain maximum efficiency. When a memory unit is called upon simultaneously to perform two tasks, the control system determines which of the two must be executed first to keep the arithmetic unit at the highest level of performance and assigns the tasks in this order.

4.1 Primary Arithmetic System

j.

The primary arithmetic system is an ultra-high speed device for performing the operations of addition, subtraction, multiplication, division and shifting. Its operation is assumed to be binary. Decimal input data is translated into binary through the use of an editing device.

The operation of the arithmetic system is checked. The system includes an indication of the location of such errors as may occur.

The numerical data words upon which the arithmetic unit operates consist of a 48-bit fraction, a 10-bit exponent, a single bit exponent over-flow indication tag and a single bit sign.

Arithmetic can be performed in fixed point, unnormalized floating point or normalized floating point notation. The accumulator is double length to facilitate double precision operations. The arithmetic unit includes four parallel registers, J, K, L, and M. The following table defines the uses made of these registers:

Operation	L	κ	L	Μ
Add	Addend		Augend Sum	
Subtract	Subtrahend		Minuend	
Multiply	Multiplicand		Multiplier Product (high	Product (low orders)
Divide	Divisor		orders) Dividend Quotient	Remainder

The arithmetic unit operates at a 10 megapulse rate. The number of cycles required for an arithmetic operation is determined by the nature of the operation and by the numbers involved.

All operations are performed in accordance with the law of signs. Negative results are expressed as true figures, rather than as complements, to facilitate direct communication between memory and the input-output units. Associated with each of the four arithmetic registers is a 10 bit exponent register, a single bit exponent overflow tag register and a single bit sign register. These four registers in combination make it possible to hold all 60 bits of a data word. During floating point arithmetic operations, exponents of the factors are directed to a separate exponent accumulator which computes the difference of the exponent and during addition and subtraction offsets the two factors when their exponents are not alike. The exponent accumulator again comes into play to adjust the exponent of the result after completion of each arithmetic operation.

The overflow tag register and the sign register also serve as selectors, and are supplemented by another selector which indicates whether the exponent exceeded its upper or lower limit.

4.2 Addition and Subtraction

Ľ.

The augend or minuend must be standing in the L register as a result of prior computation or a loading operation. The addend or subtrahend may either be in register J or may be transferred to it as a part of the add or subtract operation. For floating point operation, the difference between the exponents of the two factors is computed and the number with the lower exponent shifted to the right by the amount of the difference. The unused portion of the lesser number is shifted into the K or M register. The sum or difference appears in the L and M registers. During floating point operation, it is automatically shifted to the right or left until its highest order position lies in the highest order position of the L register. The exponent is adjusted to conform to the shift. If the number is a complement, it is converted to a true figure and a minus indication is recorded in the sign register. If the adjustment necessary to the exponent exceeds the capacity of the 10 bit exponent register, an overflow tag is recorded in the exponent overflow register to indicate that the exponent of the number has exceeded its allowable bounds. The overflow tag is carried through to the results of all following arithmetic operations involving the result as a factor to indicate that they also are incorrect.

The addend or subtrahend in register J is not disturbed by the arithmetic operation, and is available for subsequent reuse.

For double precision operations, the high order positions of the addend are entered into register J and the low order positions into register K. The augend or minuend is entered into registers L and M, and the result appears in registers L or M. In the execution of the operation, the machine will consider only the exponent and sign which it finds associated with registers J and L and will ignore any that may be associated with the numbers in registers K and M. The exponent and sign of the result will be recorded only in register L.

The actual time required to add or subtract is 0.1 microsecond. Because of the additional cycle required for switching and control, the total execution time for a fixed point addition is 0.2 microsecond. The arithmetic and control required upon the exponent in floating point arithmetic increased the time for it to 0.6 microsecond. To this must be added 0.2 microsecond for each factor or result which must be transferred to or from an arithmetic register over the bus system of the machine.

4.3 Multiplication

È

-11

When multiplying, the multiplier must be standing in register L as the result of a prior computation or loading operation. The multiplicand may either be already in register J, or transferred to it from memory as a part of the multiply operation. The product will have 95 or 96 significant positions, and will be developed across registers L and M. During floating point operation, the product will be shifted one position if necessary to place its most significant bit in the highest order position of register L, and the exponent of L adjusted accordingly.

The nominal time required for fixed or floating point multiplication is 1.2 microseconds to which must be added 0.2 microsecond for each factor or result transmitted between the arithmetic unit and any other part of the machine.

The product is automatically rounded to 48 places on conclusion of the multiplication process.

0. Q

1.4

Division

4.4

The dividend must be contained in register L prior to division. The divisor may lie in register J or may be transferred to it from memory. On conclusion of the division, the rounded quotient lies in register L and the remainder lies in register M.

The nominal time required for fixed or floating point division is 1.8 microseconds, to which must be <u>added</u> the time required to transfer factors and results between the arithmetic unit and other parts of the machine.

2.0 7

4.5 Multiple Precision

Special provisions will be included in the computer to facilitate multiple precision operations. These special provisions include the suppression of floating point exponent modification, carry from the high to the low order position of the accumulator, sign control, complementing, and rounding, none of which are allowable when performing arithmetic operations upon normal numbers.

4.6 Shifting

For shifting, the J and K registers can be treated singly or can be linked as a double length pair. The data which shifts out at the right can be discarded, or fed back in at the left. The shift can include the 48 bit fraction parts of the words, or can include all 60 bits for logical operations on whole words of data.

The data in the L and M registers can be shifted in the same manner as that in the J and K registers, with the addition that the shift may be to the left as well as to the right.

4.7 Control Decoder

The stream of instructions which is to control the operation of the computer flows into a control decoder. The decoder examines the individual instructions to determine the nature of the action which is called for. It holds an instruction which may be several instructions in advance of the one being executed by the arithmetic system. This permits it to look ahead at the program and determine what preparatory steps must be taken in anticipation of the arithmetic operations to follow. These preparatory actions include, particularly, address modification, references to memory, modification of the contents of index registers and logical transfers in the program.

4.8 Control Accumulator

A control accumulator is associated with the control decoder to perform arithmetic operations upon addresses. The control accumulator stores the index number which is to be used to modify the instruction address. The address part of an instruction which

-4.4-

is to be indexed is transferred from the control decoder to the control accumulator, where it is modified by the addition of the index number and returned to the control decoder. The control accumulator includes a tag which identifies the index register from which its contents have been obtained. If an instruction calls for an index number which is not already in the control accumulator register, the computer automatically obtains the new word from memory before performing the indexing operation.

4.9 Program Counter

1 .

A program counter is associated with the control decoder. It performs the same function as that of the program counter of earlier stored program machines. It contains the address of the instruction currently in use, and advances one for each instruction used. When a program transfer is called for, the address part of the transfer instruction is entered into the program counter. A separate bus system is provided for the transfer of data between the program counter, the control accumulator and the address area of the control decoder.



°. */

÷

PRIMARY SYSTEM

INSTRUCTION SYSTEM

IBM is engaged in an intensive program of research in the philosophy of computer control. This program is directed toward the development of instruction systems which will allow the machine to be easily programmed and which will facilitate the use of automatic programming methods. For example, the instruction system should allow the use of generalized techniques of addressing and indexing, including multiple level addressing and indexing. These techniques are expected to play an important role in future work.

Preliminary examination indicates that there are several non-conventional approaches to the problem of controlling a computer. These methods promise to make possible new programming techniques with the flexibility required for future problems.

The following list of commands illustrate the basic operations that are needed. It is not complete and is presented only as an example of the direction along which an instruction set could be developed. In particular, automatic programming methods are expected to weigh heavily in the final choice of a format. The command set will be divided into four sections. These sections are termed:

- 1. Arithmetic Commands
- 2. Register to Register Transfers
- 3. Shift Operations
- 4. Control of Operations

Selectors and Editing Operations will be covered in succeeding paragraphs.

5.1 Arithmetic Command Set

s' í.

The arithmetic instructions will control the operation of the Master Arithmetic Unit. This unit, as described previously, is composed of four registers. Each register is one full word (60 bits) in length, and has separate and specific functions to perform in the execution of instructions.

These registers carry the labels J, K, L, M. The result will always appear in the L register. It is not necessary to transfer its contents to another arithmetic register before initiating a succeeding operation.

Separate commands are available to load any register independently of others, although not all of these commands are necessary in the normal course of arithmetic calculations.

In general, an arithmetic operation will be performed on two operands, one in memory and one in register L. In actuality, register L may have to be cleared before the operation commences, in which case the operand is transferred to another register before the arithmetic operation is initiated. This transfer, when required, takes place automatically and without loss of time, and the programmer need not recognize such a case.

-5.1-

A set of basic arithmetic instructions follow. X will be taken to mean "the contents of memory location X."

1.	Add	Add the contents of X to L. The result remains in L.
2.	Subtract X	Subtract the contents of X from L. The result remains in L.
3.	Subtract L	Subtract the contents of L from X. The result remains in L.
4.	Multiply	Multiply the contents of X by that of L. Result remains in L and M.
5.	Divide X	Divide the contents of L and M by the contents of X. The quotient remains in L and the remainder in M.
6.	Divide L	Divide the contents of L by that of X. The quotient remains in L.

5.1.1 Fixed or Floating Modes

ે.

÷ 1

All of the above arithmetic instructions may be operated in fixed mode, floating point normalized mode, or floating point unnormalized mode.

In the fixed point mode, the word will be considered as occupying the S, OV, 11–58 positions of a given word or register. Positions 1–10 are ignored. Carries out of position 11 will go into the OV position.

In the floating point mode, the word is divided into a 10-bit exponent and a 48-bit fraction. The sign and overflow positions will contain the status of the respective triggers when the word is in storage, and will set those triggers when occupying a register. The register positions corresponding to the sign and overflow bits of a word will be treated as two overflow register positions. The sign and overflow conditions will be denoted by two special triggers associated with the register. These triggers will store the sign and overflow conditions of the register and their status will be stored as the sign and overflow bits of a word in storage. Only "logical" class instructions will store or load the sign overflow register positions into or out of memory.

5.1.2 Double Precision

Provision will be made for increasing the efficiency of execution of double precision operations. The exact method by which the double precision operations will be facilitated has not been decided; further study is indicated at present.

5.1.3 Absolute Arithmetic

Ţ.

;```

The following instructions will be available for absolute arithmetic:

 $|X| + L \rightarrow L$ 1. 2. X + |L|→L 3. |X| + |L|-→L 4. |X + L|-→L |X| - L -→L 5. 6. X - |L|→L 7. |X| - |L|→L |X - L|→L 8.

5.2 Register to Register Transfers

This set of commands will be divided into two groups: (1) operating with the master arithmetic unit, and (2) operating with the control arithmetic unit. Instructions within group 1 use registers J, K, L and M and memory. Within group 2, instructions use the control accumulator (CA), program counter (PC) and memory. A word in memory is divided into address parts: (1) upper (UA), (2) middle (MA), and (3) lower (LA).

5.2.1 Register to Register Transfers with Master Arithmetic Unit

- 1. Load J algebraically Enter algebraic value of X into J.
- 2. Load K algebraically Similar to 1.
- 3. Load L algebraically Similar to 1.
- 4. Load M algebraically Similar to 1.

5. Load J and reverse sign – Enter algebraic value of X into J with sign reversed.

- 6. Load K and reverse sign Similar to 5.
- 7. Load L and reverse sign Similar to 5.
- 8. Load M and reverse sign Similar to 5.

9. Load J logically - Enter value of X into J as logical bits.

- 10. Load K logically Similar to 9.
- 11. Load L logically Similar to 9.
- 12. Load M logically Similar to 9.

- 13. Load Magnitude into J - Enter magnitude of X into J with positive sign. 14. Load Magnitude into K - Similar to 13. 15. Load Magnitude into L - Similar to 13. 16. Load Magnitude into M - Similar to 13. 17. Load negated magnitude into J – Enter magnitude of X into J with negative sign. 18. Load negated magnitude into K - Similar to 17. 19. Load negated magnitude into L - Similar to 17. 20. Load negated magnitude into M - Similar to 17. 21. Load exponent into J - Enter exponent of X into J. 22. Load exponent into K - Similar to 21. 23. Load exponent into L - Similar to 21. 24. Load exponent into M - Similar to 21. 25. Load fraction into J - Enter fractional value of X into J. 26. Load fraction into K - Similar to 25. 27. Load fraction into L - Similar to 25. 28. Load fraction into M - Similar to 25. 29. Load negated fraction into J - Enter fractional value of X into J with sign reversed. 30. Load negated fraction into K - Similar to 29. 31. Load negated fraction into L - Similar to 29. 32. Load negated fraction into M - Similar to 29. 33. Store L - Enter algebraic value of L into X. 34. Store M - Similar to 33. 35. Store L logically - Enter logical value of L into L 36. Store M logically - Similar to 35. 37. Store exponent - Enter exponent of L into X. 38. Store M's exponent - Similar to 37. 39. Store Fraction - Enter fractional value of L into X. 40. Store M's fraction - Similar to 39. 41. Copy J into L - Enter value of J into L. 42. Copy J into M - Similar to 41. 43. Copy M into L - Similar to 41. 44. Copy M into J - Similar to 41. 45. Copy L into M - Similar to 41.
 - 46. Copy L into J Similar to 41.

ĵ.

 $\tau^{*} +$

	47. 48. 49.	Exchange L and M – Exchenge values of L and M. Exchange L and J – Similar to 47. Exchange M and J – Similar to 47.	
	50	Store LIA - Enter value of L in	to upper address of X.
	51	Store MA - Enter value of L into middle address of X	
	52.	Store LA - Enter value of L in	to lower address of X.
5.2.2	Regis	ster to Register Transfers with Control Arithmetic Unit	
	3	Load P.C. from II A	Entervalue of LLA of X into P.C.
	2	Load P.C. from C.A.	Enter value of 1CA into P.C.
	2.	Load P.C. from C.A.	Station to 1
	3.	Load P.C. from M.A.	Similar to 1.
	4.	Load P.C. from L.A.	Similar to I.
	5.	Store P.C. in U.A.	Enter value of P.C. in U.A. of X.
	6.	Store P.C. in CA	Enter value of P.C. in CA.
	7.	Store P.C. in M.A.	Similar to 5.
	8.	Store P.C. in L.A.	Similar to 5.
	•••		
	9.	Exchange P.C. and U.A.	Exchange values of U.A. in X and P.C.
	10.	Exchange P.C. and CA	Exchange values of CA and P.C.
	11.	Exchange P.C. and M.A.	Similar to 9.
	12	Exchange P.C. and L.A.	Similar to 9.
	13.	Load CA from U.A.	Enter value of U.A. in X into CA.
	14.	Load CA from M.A.	Similar to 13.
	15.	Load CA from L. A.	Similar to 13.
	16.	Load CA from U.A. with	Enter value of U.A. in X into CA with
		sign reversed.	sign reversed.
	17.	Load CA from M.A. with	•
		sign reversed.	Similar to 16.
	18	Load CA from L.A. with	
		sion reversed	Similar to 16.
		sign reversed.	
	19.	Load CA with magnitude	Enter magnitude of U.A. in X into CA.
		of U.A.	
	20.	Load CA with magnitude	
		of M.A.	Similar to 19.
	21.	Load CA with magnitude	
		of L.A.	Similar to 19.
	00		
	22.	STORE CA IN U.A.	Enter value of CA Into U.A. of X.
	23.	Store CA in M.A.	Similar to 22.
	24.	Store CA in L.A.	Similar to 22.
	25.	Store CA magnitude in U.A.	Enter magnitude of CA into U.A. of X.

-5.5-

÷ È

26.	Store CA magnitude in M.A.	Similar to 25.
27.	Store CA magnitude in L.A.	Similar to 25.

5.3 Shift Operations

<u>_</u>___

1 1

Shift operations are used to move the bits in a word to the right or left of their original positions in the L or M register, or both. With the exception of the ring shift instructions, zeros are automatically introduced into the vacated positions of a register. Logical shifts will be defined as shifts which treat the sign position in the same fashion as other bits in the word. In algebraic shifts, shifting is equivalent to multiplying by a power of 2. Ring shifting will enter bits shifted off of one end of a register into the other end of the register. Instructions in this set are listed below.

- 1. Shift L register left.
- 2. Shift L fraction left.
- 3. Shift L logically left.
- 4. Shift L register right.
- 5. Shift L fraction right.
- 6. Shift L logically right.
- 7. Shift L and M left.
- 8. Shift L and M fraction left.
- 9. Shift L and M logically left.
- 10. Shift L and M right.
- 11. Shift L and M fraction right.
- 12. Shift L and M logically right.
- 13. Ring shift L left.
- 14. Ring shift L fraction left.
- 15. Ring shift L logically left.

16. Ring shift L right.

- 17. Ring shift L fraction right.
- 18. Ring shift L logically right.
- 19. Ring shift L and M left.
- 20. Ring shift L and M fraction left.
- 21. Ring shift L and M logically left.
- 22. Ring shift L and Mright.
- 23. Ring shift L and M fraction right.
- 24. Ring shift L and M logically right.

5.4 Control Operations

÷.

- ¹ (

The control instructions in this section perform two functions - (1) modify the contents of index registers and control accumulator and, (2) direct the course of the program. Instructions in the latter group may be an unconditional direction or a conditional one. Conditional transfers in this section are concerned with transfers based on the status of index registers of the CA. Conditional transfer instructions concerned with the status of the master arithmetic unit, selectors or memory are not considered.

In referring to an index register, the contents are treated as three entities – (1) Current value (C), (2) Test value (T), and (3) Modifier (M). An index register will be indicated by the letter 1.

1.	Transfer and Set Index	Store the program counter in I and transfer to X.
2.	Transfer on zero CA	If CA is zero, transfer to X. If not, proceed to next instruction.
3.	Transfer on non-zero CA	If CA is zero, proceed to next instruc- tion. If not, transfer to X.
4.	Transfer on CA overflow	If CA overflow indicator on, transfer to X. If not, proceed to next instruction.
5.	Transfer on no CA overflow	If CA overflow indicator on, proceed to next instruction. If not, transfer to X.
6.	Transfer on CA plus	If CA is plus, transfer to X. If minus, proceed to next instruction.
7.	Transfer on CA minus	If CA is minus transfer to X. If plus, proceed to next instruction.
8.	Add PC and Transfer	Add program counter to CA and transfer to X.
9.	Transfer on Index	In index register I, add M to C. If new C less than T, transfer to X. If C greater than or equal to T, proceed to next instruction.
10.	Transfer on No Index	In index register I, add M to C. If new C greater than or equal to T, transfer to X. If C less than T, proceed to next instruction.

۰.			
J	11.	Transfer on Low Index	In index register 1, if C less than T, transfer to X. If C greater than or equal to T, proceed to next instruction.
	12.	Transfer on High Index	In index register I, if C greater than or equal to T, transfer to X. If C less than T, proceed to next instruction.
	13.	Transfer on Index by CA	Add the contents of the CA to C in index register. If new C less than T, transfer to X. If greater than or equal to T, proceed to next instruction.
	14.	Transfer on No Index by CA	Add the contents of the CA to C in index register 1. If new C greater than or equal to T, transfer to X. If C less than T, proceed to next instruction.
	15.	Transfer on Index rel. to CA	In index register, add M to C. If new C less than CA, transfer to X. If greater than or equal to CA, proceed to next instruction.
/	16.	Transfer on No Index relative to CA	In index register I, add M to C. If new C greater than or equal to CA, transfer to X. If less than CA, proceed to next instruction.
	17.	Transfer on Low Index relative to CA.	If C of index register I less than CA, transfer to X. If C greater than or equal to CA, proceed to next instruction.
	18.	Transfer on High Index relative to CA	If C of index register I greater than or equal to CA, transfer to X. If C less than CA, proceed to next instruction.
	19.	Index	In index register I, add M to C.
	20.	Negative Index	In index register I, subtract M from C.
	21.	Index by CA	Add contents of CA to C of index register 1.
	22.	Negative Index by CA	Subtract the contents of CA from C of index register 1.
/	23.	Shift CA left 	Shift the CA left X places. Zeros are inserted in the vacated positions and bits shifted off the left end of the CA

		are lost. If a one is passed through the overflow position, the CA overflow indicator is turned on.
24.	Shift CA right	Shift the CA right X places. Zeros are inserted in the vacated positions and bits shifted off the right end of the CA are lost. No CA overflow is possible.
25.	Add to CA	Add the contents of X to the contents of the CA.
26.	Subtract from CA	Subtract the contents of X from the contents of the CA.

35 A

.

SELECTORS

The machine contains a set of electronic selectors, each of which may be set either off or on, as denoted by a binary 0 or 1. Each selector may be addressed numerically as one of a series of 256 units. It is also possible to read the settings of the selectors in groups of 60 as data words which can subsequently be subjected to logical manipulation, and to set the selectors in block of 60 as logical data words.

6.1 Selectors

-<u>,</u>i,

ŝ

Included among the selectors which will be provided are the following:

Selector	Function
1-60	Program Set Selectors
61-70	Manual Set Selectors
71	Register L is minus
72	Register L is zero
73	Register M is minus
74	Register M is zero
75	Register J is minus
76	Register J is zero
77	Control Acc is minus
78	Control Acc is zero
79	Control Acc High
80	Control Acc Equal
81	Exponent of result exceeds maximum negative
82	Exponent of result exceeds maximum positive
83	The fraction part of the result has overflowed (occurs only in fixed point operations)

Selector	Function
84-90	The I/O unit assigned to this selector is in operation
91	Same I/O unit requires special attention
92	Register M overflow condition
93	Register K overflow condition

The Program Set Selectors #1-60 can be set up by program control and can be tested by it to control further action. Each provides a light on the control console, and a switch which when set on causes the machine to stop if the selector is any time set on by the program.

These selectors may also be addressed from a register as a block of 60 selectors. Each bit position of the register will designate for examination the respective Program Set Selector and by means of special commands, these selectors may be operated upon in any desirable combination.

The Manually Set Selectors #61-70 are on the control console. They may be tested by the program to direct it in any desired manner.

Selectors #71–78 record the current status of the sign triggers associated with the various arithmetic and control registers in the machine and may be tested by the program.

Selectors #79–80 are set on the basis of comparisons made with the contents of the control arithmetic unit.

Selectors #81-82 are set automatically whenever the exponent of a result exceeds the permissible limits. They are also provided with a light and a reset push button on the console. When the selectors are set on, they remain on until reset by the program or the manual reset button.

Selector #83 indicates the setting of the overflow trigger of the L register. This trigger is set by the contents of the overflow bit position of the word in the L register.

6.2 Selector Instructions

<u>, ^</u>,

, h,

The following instructions will be available in order to operate on selectors:

1. Transfer Z on Transfer to X if selector Z is on

2. Transfer Z off	Transfer to X if selector Z is off
3. Logical Transfer	Transfer to X if any of the terms in the logical expression is true.
4. Parallel Transfer on	Transfer if all selectors correspond- ing to one bits in L register are on.
5. Parallel Transfer off	Transfer if all selectors corresponding to one bits in the L register are off.

(In instructions 4 and 5, ignore selectors corresponding to zero bits in the L register)

6.	Set Z on	Set selector Z on
7.	Set Z off	Set selector Z off
8.	Invert Z	Invert the setting of selector Z
9.	Parallel Set on	Set selectors "on" corresponding to one bits in L register.
10.	Parallel Set off	Set selectors "off" corresponding to one bits in L register.
11.	Parallel Invert	Invert the settings of the selectors addressed by one bits in the L register.

6.3 Logical Connectives

.5

J.

Provision will be made for applying any binary logical connective to two arrays of 60 bits to produce a third 60 bit result. All of the 16 possible connectives except the two which are trivial will be provided. These connectives provide an important new tool for logical control, and will sometimes permit a single instruction to be the equivalent of as many as sixty logical instructions of more conventional form.

6.4 Real Time Input-Output

Two sixty point plug connectors will be provided, one serving as input to the computer and the other as an output. The connectors can be used by the customer to attach real time devices to the machine. The sixty point connections can be treated as data words by the computer, or the sixty individual connections can be addressed individually as selectors for control of the computer or computer control of the real time device.

6.5 Example of Use

Å.

г¹,

Provision will be made in this computer to exploit the logical power of selectors in a manner hitherto unachieved. An example will serve to indicate the ease with which selectors may be used in this machine.

Suppose that the following logical expression is to be tested for validity:

 $\overrightarrow{AB} \vee \overrightarrow{ABC} \vee \overrightarrow{AC}$

Where A, B, C are selector numbers. In order to specify the validity test of this expression, the following information is necessary:

- 1) Operation code (Logical Transfer)
- 2) Selector numbers (A = 3, B = 8, C = 2 in the above example)
- 3) Codes for the logical connectives:

0 = end of expression 1 = AND 2 = OR $3 = \overline{Y} AND (not Y AND ---) \qquad Y is any$ $4 = \overline{Y} OR (not Y OR ----) \qquad selector number$

4) Transfer address (for example, location 312)

Using the codes given in 3 and the given selector numbers, the expression

is translated into

3, 1; 8, 4; 3, 1; 8, 1; 2, 4; 3, 3; 2, 0

The commas and semicolons would not appear in machine language and are given here to show the divisions of the expression. Each pair of numbers between semicolons denotes a selector number and the connective following it. The zero indicates the end of the expression. Since there are no limits on the length of terms or the length of expressions, it is apparent that any logical expression which can be reduced to the Alternational Normal Form can be evaluated, and any logical expression may be so reduced. In this example, the machine will examine the encoded expression from left to right and will transfer to location 312 as soon as any term in the expression results in a "valid" or "true" indication. As soon as a valid indication is detected, the examination ceases and control is transferred to the address stated in the "Logical Transfer" instruction. Often, the numerically encoded form of the expression will occupy more than one word; in fact, it may occupy as many consecutive words as are necessary to contain it. The machine will automatically examine the expression term by term until it either indicates a valid term or reaches an "end of expression" code. Any number of terms of any size may be encoded and it is not necessary to recognize word divisions in the encoding.

÷.

A.

Thus, the testing of any combinations of selectors may be indicated by direct translation of logical expressions using the standard notation of symbolic logic. This examination will proceed at a very high rate of speed.

EDITING OPERATIONS

The process of mapping information from one language, code, or form into another is usually called "editing." Examination suggests that the table lookup method of transforming information from one form into another is a very fundamental approach that has been overlooked in the past. Table lookup has the advantage that it can represent any relation that can be reduced to the form of a one-to-one correspondence. If special provision is made for automatically sequencing the table references, many different mapping functions can be performed on information at a very high rate of speed. To this extent, the general method of editing in this computer will be based upon a special form of table lookup command which will allow considerable variation to be introduced into the basic table lookup process.

Thus, provision can be made for general methods of handling editing operations in a rapid and automatic manner. In particular, the editing associated with input/output operations will be implemented through special commands which will automatically transform bit codes from any code to any other code for which a one-one or many-one relation can be established. For example, this will apply to the conversion of a 6-bit code to a 4-bit code or vice versa. An example of this case will be described later. These special commands will also include the case of converting any character code of N bits/character to any other character code of M bits/character, where N and M are not equal.

The case of number conversion from one radix to another radix will be provided for in a manner which allows the conversion to proceed without interrupting the work being done in the primary arithmetic unit. The input/output computer will contain both a decimal and a binary arithmetic unit to aid in radix conversion.

The expansion or contraction of the data format is often necessary as a part of input/ output operations. Provision for very flexible control of this operation will be furnished as a part of the special commands provided for editing operations.

It is intended that the editing capabilities of this computer will be general enough to handle almost any variation in form without undue effort in programming.

7.1 Editing Example

<u>, </u>

şŝ,

A simple example of editing will serve to indicate the flexibility of the editing system. This example is chosen for its simplicity of presentation.

The problem: convert 132 characters of numerical data written in an arbitrary 6-bit code to an arbitrary 4-bit code. This conversion will be accomplished by the use of a 10 word table, each entry of which contains the 4-bit character to be used as a replacement for any given 6-bit character in the original data. It is obvious that the table lookup procedure to be used is of a standard logical nature.

-7.1-

To accomplish this in automatic manner, the following information must be specified:

1. Number of bits per data character (6 bits in the example above).

Δ.

А

- 2. Number of bits per converted character (4 bits in the example above).
- 3. Total number of characters to be converted (132 in this example).
- 4. Starting address of the 6 bit data in memory (location 12,000 in this example).
- 5. Starting address for storing converted 4-bit data (location 13,000 in this example).

Once the above information has been given, the conversion proceeds automatically and without any further programmed assistance until 132 characters of 6 bits each starting at location 12,000 have been converted to 132 characters of 4 bits each and stored beginning at location 13,000.

Since the number of bits in each character can be specified, it is apparent that the above conversion could be effected between any two character codes that can be specified in the instructions.

INPUT AND OUTPUT

The input-output section is an integrated communication system for the transfer of information between the input-output devices of the computer and its memory, and between the input and output devices themselves. The operation of the input-output system may be from a manual console or it may be completely under computer control. It is designed to adapt itself to accept and control a wide range of input-output devices, including the following:

- Type 727 Magnetic Tape a)
- b) Higher Speed Magnetic Tape
- c) Magnetic Disc Memory
- d) Card Reader
- Card Punch
- Electronic Printer-Plotter
- e) f) g) h) 1000 line-a-minute Mechanical Printer
- **Control Typewriter**

Many other input-output devices not in the above list can be used provided appropriate adaptive measures are taken with these devices. In addition, the computer can accommodate data transmission systems used with instruments and control devices at remote points. (see Selector System)

The input-output includes an arithmetic and control system. This mechanism is in effect a computer designed to compute memory addresses for input-output data and to edit data for later use by the primary computer section of the machine. The input-output part of the machine is shown by the accompanying diagram.

8.1 Modes of Operation

ė,

A.

The input-output units will be capable of operation in the following manner:

- 1) Each unit can communicate with the memories of the computer to handle data with which the primary computer system is concerned. This data can be edited or otherwise processed by the inputoutput computer.
- 2) Each unit can communicate with the memories of the computer and thence with any other inputoutput unit. While in memory, the data can be processed by the input-output computer to change machine codes, edit for printing, etc.

-8.1-

3) Direct communication will be possible between a Type 727 Tape Unit and a card reader, printer or punch to provide the equivalent of the present auxiliary equipment.

The above three modes of operation are controllable either through the input-output computer console or the computer.

8.2 Input-Output Interchanger

<u>.</u>`.

J,

The input-output interchanger provides the communication link between the individual input-output devices and the magnetic core memories. The interchanger provides a single word register for each input-output device which is in operation at any given time. When connected to an input device, this register transfers data to memory one word at a time as it becomes available from the input mechanism. In a similar manner, the register transfers information a word at a time from memory to an output device. As information is transferred between the input-output device and the interchanger register, a translation is made between the sixty bit parallel representation of the memory, and the more serial representation employed by the input-output unit to which the interchanger register is connected.

8.3 Input-Output Computer

The input-output computer includes four arithmetic registers. These registers communicate in parallel with the computer memories and serially with devices in the input-output system providing the following functions:

- a) a binary adder
- b) a decimal adder
- c) a logical connective system

These registers and their connections make it possible to add, subtract, multiply and divide in both binary and decimal notation, as well as to perform complex logical and editing operations. The input-output computer will be provided with an instruction decoder and program counter to permit the program on which it operates to be independent of that of the primary computer.

8.4 Input-Output Addressing

An input-output address register computes the memory address of each new word sent between an interchanger register and the computer memory. For this purpose two words of data will be held in register storage for each input-output unit which is in operation at a given time. These words will be divided as one-half words to record the following:

- The address in memory at which the first word of the block of data associated with the input-output unit is to be stored.
- 2) The address of the next word to be handled.
- 3) The address of the final word.
- 4) The address of the instruction to be used following completion of the input-output operation. This address is required only for special control operations.

As each new word is communicated between an interchange register and the memory, a signal is given which refers the computer to the registers for the input-output unit to obtain the next address to be used. After being used, the address is increased by one and stored. At the same time, the address is compared with the address of the final word in the block to determine whether or not the input-output operation has been complete. If so, it can signal the beginning of a program sequence. The starting address is retained in case it should be necessary to restart the unit, as might be the case in the event of a reading error on tape.

8.5 Input-Output Multiplexing

2.

The input-output system is designed to operate a number of input-output units simultaneously. To accomplish this, there must be as many interchange registers as there may be input-output units in operation at any one time. The number of these registers can be varied to suit requirements. After completion of an inputoutput operation, an interchange register becomes available for reassignment to whatever other unit may require it next.

Two bus systems are provided between the interchange registers and the individual input-output units. One of these connects the disc memories to the interchange registers. The communication channel is parallel because of the 15,000,000 bit information rate. All other input-output units connect through a more serial bus system to the interchange registers. Both channel systems are multiplexed to permit each of them to be used by a number of input-output units.

A priority system will establish, on the basis of relative need, the order in which the bus systems will be used by the individual input-output unit to communicate with the interchange registers. A similar system will establish whether the primary computer or the input-output system will have priority in making a memory reference to a unit required by both. The input-output reference will be deferred unless it is urgently needed.

8.6 Input-Output Console

The input-output system will be provided with its own control console, which will provide manual control over the input-output computer and the assignments of the individual input-output units. This manual control will supplement the fully auto-

matic control of the input-output which is available through the computer program.

8.7 Tape Units

 \mathbf{r}_{λ}^{i}

+

The present IBM Type 727 Tape Units will be available for use as input, output and intermediate storage. These units are a 1/2" wide and 2400' long plastic tape coated with magnetic oxide. Seven tracks are recorded on the tape, of which six carry data. The seventh is used for checking. The recording density is 200 bits to the inch. The linear speed is 75" a second. The information rate is 90,000 bits a second.

IBM is presently developing higher speed magnetic tape units. The computer will be adaptable to the attachment of these tape units as they are developed.

8.8 Magnetic Disc Memory

Magnetic disc memory devices provide an additional reservoir for instructions and data. This memory is available in modules of 1,048,576 words of data. This data can be transferred between the magnetic disc unit and the magnetic core memory at the rate of 15,000,000 information bits a second.

The flow of information will be continuous from consecutive blocks. Random access to any block of information will require up to 100 milliseconds, depending upon the address.

If more than one disc memory is attached to the computer, it will be possible to communicate with two or more at the same time.

Checking will be provided for the disc memories. The track selection and addressing system will be checked.

8.9 Electronic Printer-Plotter

An electronic printer and point plotter will be available. This machine will record on 35 millimeter microfilm at the rate of 16,500 characters a second. The film record can subsequently be transferred to paper to provide one or as many copies of each page as is desired. A secondary display tube can be provided for operator surveillance. The printer can record 64 characters, including 47 characters of the Type 407 Accounting Machine together with symbols especially suited to point plotting.

When used for page printing, the electronic printer will print 128 characters in a line and 64 lines on a page. The time required for each page will depend primarily upon the amount of data to be recorded. The printer will be free to move from any part of the page to any other between any one character and the next.

When used as a point plotter, the electronic printer will provide a 1024×1024 raster.

The control will allow the following:

- 1) Single recording of a display word.
- 2) Automatic recycling to display the point called for by the control word until a new word is introduced into the Interchange register.
- 3) Automatic ordinate generation.
- 4) Automatic abscissa generation.
- 5) Automatic intensification of any desired point to facilitate interpretation.

The first 8 columns on the left side of the film can be set aside for coded control data.

8.10 Mechanical Printer

• •

 \mathcal{F}_{i}

The computer can be equipped with a directly connected mechanical line printer. This printer will provide 120 columns of data, and will operate at the rate of 1,000 lines a minute.

The horizontal column spacing will be 10 to the inch, and the vertical spacing 6 to the inch. Each printing position will be capable of recording any of the 47 characters presently in use in the Type 407 Accounting Machine.

The printer will be equipped with an automatic tape operated carriage. Traction devices above and below the printing line will provide positive registration and form feeding.

Associated with the printer is a 120 character buffer into which each line to be printed is automatically transferred. The data is received from the memory of the computer through an output interchange register.

It is checked by carrying forward with each word the check bit which was stored with it in memory. These bits are checked against the actual operation of the print mechanism to insure that the electrical and mechanical action as well as the data transfer has been properly executed. It may later be proposed that a <u>magnetic printer</u> operating at the same or higher speed be substituted for the mechanical printer.

-8.5-

8.11 Card Reader

1.

а.

The card reader operates at 500 cards a minute. It includes a 12 x 80 card image buffer storage, which makes it possible to read cards which have been coded in either Hollerith or binary fashion. Each 5 columns of the card will be transferred through the interchange register to memory as a single 60-bit word. The reading of the card will be checked. The transfer of information to memory will be checked.

The card reader will include a <u>control panel</u>, which will permit the rearrangement of the order of the card columns and the deletion of unwanted columns. The speed of the card reader may be increased to 1000 cards a minute during this program.

8.12 Card Punch

The card punch will operate at 155 cards a minute. It will include a 12×80 card image buffer capable of recording in the card in either Hollerith or binary notation. The transfer of data from memory through the output interchange register to the punch, and the electrical and mechanical operation of the punch itself will be checked.

The card punch will include a <u>plugboard</u> which can be used to rearrange the order in which the columns of data are punched or to prevent the punching of any desired columns.

8.13 Interrogation Typewriter

An input-output typewriter is provided as a part of the operator's console. It can be used for the manual insertion of instructions or data as well as printing from memory at the rate of 600 characters a minute.

Data is transferred between the typewriter and memory through an interchange register in such a way that the operation of the computer is not held up by the typewriter. The transfer of data to and from the typewriter and the mechanical operation of the typewriter itself are checked.



INPUT - OUTPUT SYSTEM

€

7.

· -...

ERROR DETECTION AND CORRECTION

As a part of the program to achieve unattended operation of the computer, important new steps will be taken in the design to insure a low incidence of errors and a minimum interruption of productive work for maintenance. These steps will include automatic error detection throughout, automatic single error correction in large parts of the machine, marginal checking to anticipate future troubles during regular scheduled maintenance, automatic error location of many classes of troubles, and pluggable modular design to permit the rapid replacement of faulty parts.

9.1 Automatic Checking

ι, .

A,

The automatic checking system will be extended throughout the machine. The check will be such as to detect single errors and most multiple errors. The manner in which checking will be accomplished will vary with the nature of the circuitry being proved.

9.2 Automatic Error Correction

Automatic error correction will be provided in many areas of the machine. The methods used in these areas will be one which will detect all single and double errors and most multiple errors, and will correct single errors. This will require the inclusion of additional bits with each sixty-bit word for detection and correction. As the design proceeds, an effort will be made to include automatic error correction in whatever areas of the machine it is desirable. Within those parts of the machine to which autocorrection applies, the incidence of occasional single bit errors need not affect the over-all performance of the machine. The process of automatic error correction will facilitate the location of the bit in error and the subsequent replacement of the unit at fault.

The arithmetic unit will be arranged to <u>repeat the basic arithmetic operations of</u> addition, subtraction, multiplication, and division if an error is found during their execution.

Whenever an error is detected and corrected, the machine can record the circumstances of the error. This information will be available to the engineer for the later location and correction of the component which is at fault.

ENGINEERING

In order to achieve the ultimate in performance in the computer, all aspects of the system must be brought to the same high engineering level. The system will be engineered as a unified whole, with all parts of the machine being compatible, and representing the best of the art in that area. To this end, developments in all areas of the computer field are being considered in the specification and design of this machine. IBM is conducting research and development in all major areas of the computer field, with special emphasis on those techniques and components which will give the greatest increase in overall performance and reliability in the finished machine.

The performance specifications described in the preceding sections are based on the results of the research and development work carried on at IBM, as well as on the experience derived from present IBM products. It is felt that the predicted performance of the machine, while representing bold technical advances, is based on sound engineering results presently being obtained in the laboratory. All components and techniques to be used in the machine will be subjected to critical examination and test before they are accepted for final use. The special characteristics of this machine will strongly influence final choices.

10.1 Automation of Design

The IBM Engineering Laboratory has a computing bureau which has at its disposal the latest IBM computing equipment. Intensive use will be made of this computing facility in the design and engineering of the machine components and system. The use of machine computation in the design of the circuits and logic of this computer will decrease the engineering man-hours required and yield a final design which is much nearer the optimum. Some areas in which the laboratory computing facilities will be used are the following:

- Basic circuits design. Machine computation permits a much more thorough and α. accurate analysis of the individual basic circuits in the computer. The analysis includes the variation of the performance of the circuits as operation conditions and components vary. A more accurate prediction of the margins and performance of the completed system is possible when machine computation is used.
- b. Optimization of logical design. The logical structure of large computers is so complex that it is not feasible for the designers to evaluate all the possible alternative designs. The use of the IBM 704 makes it possible to postulate and evaluate several different logical configurations which achieve the desired result, and to select the best of these alternatives.
- System simulation. The behavior and ease of use of the system can be evaluated с. by simulation of the new machine on existing computers. This technique has proven of great value in the design of the present line of IBM equipment.

-10.1-

- d. Optimum physical layout. The use of topological programs permits the generation of optimum layouts of circuits on machine panels. Previously, the layout of the machine logic in terms of physical circuits on machine frames was governed by experience and intuition on the part of the individual engineer. The necessity of very short signal leads imposed by the extreme speed of the machine virtually precludes this technique.
- e. Engineering records. Machine accounting techniques applied to the engineering records and change procedures during the designing and construction of the computer will reduce the engineering man-hours required, and provide accurate reliable records.

10.2 Reliability

 i_{i}

÷.

Reliability will be emphasized in all phases of the design of the computer. The objectives of the reliability program are threefold:

- a. To obtain unattended operation with periodic servicing.
- b. To obtain maximum error-free operation time.
- c. To minimize preventive maintenance service time and fault correction times.

Conservative engineering, carefully selected and controlled components as well as automatic checking and extensive fault prediction techniques are the techniques which will be used to achieve these objectives.

10.2.1 Components

"Computer quality" components will be used throughout. It has been IBM's experience that high quality components can be obtained and must be used in electronic computers. In addition, it is obvious that as the size, i.e. number of components, of a computer increases, the reliability of the components must increase by a greater factor. Several factors enter into obtaining reliability in a component. These are discussed in the following paragraphs.

Specifications

The specifications must be based on the ultimate use of the component, and must control the characteristics which affect the performance of the component in the final circuit. The specifications can be arrived at only after extensive tests, including life tests, have been made on the component, and the application has been studied. The specifications must describe a component which can be manufactured with a reasonable yield, and must control its characteristics in such a way that it will have reliability and long life in the final circuit. In general, the presently existing MIL specifications and commercial specifications either do not control the characteristics which are important to the computer designer, or are inadequate in the amount of control. For this reason, IBM has found it necessary to write its own specifications for components for use in computers. Many of these specifications have since formed the basis for industry-wide specifications for these same computer components. The work done by the IBM Quality Control organization and the analysis of field returns by the Electrical Laboratory have verified the worth of these specifications. As new applications and new components are used, it is necessary to write specifications covering these applications and components.

Manufacturing procedures and controls

Not only must the raw materials which go into a component be high quality, but the manufacturing processes and quality control techniques which are used must be carefully spelled out. It has been IBM's experience that techniques and controls which are general practice in the industry, and which may even conform to MIL specifications are frequently not adequate for computer applications. Procedures which are now in use do not necessarily control the processes or tend to produce quality in the areas which are best for computer applications. For example, controlling the small signal characteristics of transistors is not necessarily the correct thing to do if the transistors are to be used in switching circuits which drive the device from saturation to cut-off.

Inspection and tests

e,

Before any component is used in the construction of the machine, it must pass initial acceptance tests on all characteristics which are important in the application for which it is intended. Where applicable, lot sample life tests are conducted on the components. In the case of new components which have not been used in previous IBM equipment, and for which no field experience exists, extensive life tests are conducted so that the component may be evaluated, and if found acceptable, suitable design limits may be established and specifications written.

Design data

The information necessary to the circuit design engineer is obtained from extensive tests, life tests and previous experience. This data is gathered by the components group and is not left to the individual circuit designer. If the circuit engineer must use some characteristic of a component which is not controlled by the specifications, it must be established that the characteristic has a good correlation with presently controlled characteristics, or the specification must be rewritten to control that characteristic.

Component control

`./

٠,

The component applications group is given control over use of each component in the final circuits. This control assures that the component is used as it should be to obtain full benefits from all the controls and quality which have been built into the component. This removes the responsibility from the individual circuit designer, who may be under pressure to misuse the component and puts it in a central authority whose sole responsibility is to see that the greatest reliability may be achieved from the components. The component control group is the central clearing house for information pertaining to the failure of components, and serves as a check on the incoming inspection, purchasing and the vendor's manufacturing activities. In brief, the group has the final responsibility for the reliability of the components in the machine, watching over the application of the components, and the quality of the incoming components.

Components manufactured by IBM

In many previous instances it has been necessary for IBM to manufacture its own components to achieve the quality and reliability that are necessary for a computer application. One instance was the development and manufacture of the special cathode-ray tube for use in the 701 and 702 memories. Another case is the manufacture of ferrite memory cores for use in the 704, 705 and other applications. The majority, if not all of the transistors to be used in this machine will be of IBM design and manufacture. These devices will be manufactured under highly controlled conditions and to specifications which are precisely those used in the design of the circuits employing the transistors. During the development of these transistors, emphasis will be placed on achieving reliability in the final product, and continual work will be conducted to evaluate and improve the reliability. Constant testing and evaluation of the transistors as they are developed will give the circuit designers detailed information as to the significant parameters which affect the design of their circuits, and they in turn can supply the transistor designers with information as to which characteristics must be given special attention and controlled the most carefully.

It is believed that the above procedure, coupled with careful and intelligent manufacturing techniques, good quality control procedures and life tests can result in reliable performance from a relatively new device.

Purchased components

As has been the case in previous machines designed and manufactured by IBM, many of the components used in the construction of the machine will be purchased from outside sources. In order to achieve the necessary reliability in these components several fundamental rules must be followed carefully. These rules are as follows:

- a. Only those components which, in the judgment of experienced people, show the most promise of ultimate reliability will be chosen for use in the computer.
- b. The best vendor of these components will be selected on the basis of past performance, plant facilities, ability, and willingness to meet specifications.
- c. Stringent but realistic specifications will be written for each component as described above.
- d. Vendor Education and Cooperation

It has been our experience in the past that one of the requisites of a good source of reliable components is an informed and cooperative vendor. One of the criteria for selecting vendors in an indicated willingness to cooperate in the required special manufacturing techniques, quality control techniques and purchase specifications. Whereas many of the components used in a computer are very much like those which are standard in television, the entertainment industry and the Military, the application, specifications and performance requirements are usually much different. Hence, the manufacturer must be educated as to the needs of the particular application, and must be willing to cooperate to the fullest in the maintenance of the over-all quality and uniformity of the components.

10.3 Circuit Design

<u>д</u>,

٩

The circuits in the machine will be designed to give the highest reliability. Most of the machine will be composed of a small number of standard circuits which will be the basic building blocks from which the logic will be constructed. These circuits will be carefully engineered and exhaustively tested for satisfactory operating characteristics.

The responsibility for the ultimate reliability of the circuits will reside in a circuits control group. This group will design some of the circuits, and will approve all circuits that are used in the machine. The approval of this group will mean that the circuits have met the following criteria:

- a. No component is used in such a way as to be detrimental to its useful life. Approval of the components control group is required.
- b. An "end of life" value will be established for all components. Any circuit will continue to function properly when any one of the components has drifted to this predetermined "end of life" value.
- c. Proper allowance has been made for the initial purchase specification tolerance in all components in the circuit.

- d. A means is provided for detecting the drift of a component that is approaching the "end of life" value, or for detecting a combined drift of several components which will effect the operation of the circuit. In other words, each circuit must have a means of easy and accurate marginal checking to detect impending trouble before such a trouble occurs.
- e. The circuit is, in general, well designed and does not use components in such a way that the purchase specifications and quality control procedures will not assure a high quality component for the particular usage.

10.4 Modular Design

<u>,</u>"-

٩.

The design of the component circuits will be modular. A small number of different types of pluggable units will make up a large part of the machine. Due to the parallel nature of the machine, many parts will be similar. In addition, many functions are repeated throughout the machine and will use interchangeable pluggable units. For instance, many of the registers of the machine will be similar and will be composed of identical modular elements. The same applies to the parallel switching mechanisms for communication over the bus system. Wherever one unit may differ from another in some minor respect, an effort will be made to design a single unit which can be used in an emergency in both places even though it may not provide the least number of components. Such units will therefore be suitable for general use at either point.

The logic of the machine will be fitted into modules in such a way that trouble location is facilitated. The trouble need only be localized to a module, and not to the specific circuit or component.

MAINTENANCE

It will be possible to do nearly all maintenance during regularly scheduled maintenance periods. During these periods, the emphasis will be on the rapid location and correction of impending troubles. To do this, rapid and accurate trouble prediction facilities will be provided.

11.1 Trouble Prediction and Location

Facilities will be included to assist in the location of impending troubles so that these can be corrected during the preventive maintenance period to permit unattended operation of the computer.

11.2 Marginal Checking

Ć

The computer will include extensive marginal testing equipment. This equipment will, in general, vary the operating voltages of the various parts of the machine above and below their normal values in order to isolate individual components which may be drifting in the direction of failure. This will be done during the regularly scheduled maintenance periods for the machine and will help to guarantee against interruption of scheduled operating periods. In certain areas of the machine, the marginal checking mechanism will be continuous and automatic. The marginal test voltage lines will be divided into sections corresponding to the logical functions of the machine. This division facilitates the location of troubles. The detail of the logical division of the lines will be under control of the operator. Provision will be made for automatic selection and testing of marginal testing lines.

11.3 Monitor Circuits

In certain areas, special monitor circuits will be provided to check the adequacy of the operating circuits, and detect impending failure. These monitors will augment the marginal checking system.

11.4 Error Location Circuits

As pointed out in Section 9, automatic error detection will be provided in the principal information paths of the machine. The error detection circuits will have additional equipment to permit the location of the error within a specific area.

A combination of the above techniques should permit the detection of most impending troubles before they cause a machine malfunction during normal operation. Experience has shown that no single technique is completely effective, as each has advantages and disadvantages. The proper application of all these techniques to the appropriate parts of the machine will result in much improved trouble prediction and should give extended periods of unattended operation.

-11.1-

5 Trouble Correction

Once the trouble or impending trouble has been located, it must be repaired quickly and completely. Provision must be made to bring the system back to a reliable condition with adequate margins. This will be accomplished by removing the module of the machine at fault and replacing it with a spare which has been thoroughly tested for operating margins.

After the defective module has been removed, it can be tested and the particular component or components at fault located and replaced. The unit will then be tested for adequate operating margins, inspected for quality of mechanical parts and replaced in spare stock. If the unit is reconditioned and passes the marginal tests, it can be placed in operation with assurance that the module will function properly in the machine.