

**REFERENCE
MANUAL**

LGP-21

COMPUTER SYSTEM

**COMMERCIAL COMPUTER DIVISION /
GENERAL PRECISION, INC**

LGP-21 COMPUTER SYSTEM

REFERENCE MANUAL

Preliminary Version

This manual is intended as a basic introduction to the LGP-21 computer system. It summarizes all LGP-21 hardware features, operating procedures and programming techniques. A more complete treatment of these topics will be found in the LGP-21 programming manual.

CONTENTS

LGP-21 COMPUTER SYSTEM

PAGE

GENERAL CHARACTERISTICS	1
Input/Output Device	2
Word Structure	2
MEMORY	3
COMPUTER CONTROL	4
Accumulator	5
Instruction Register	5
Counter Register	5
COMMAND LIST	5
The Command List	5
A m ADD	5
B m BRING	6
C m CLEAR	6
D m DIVIDE	6
E m EXTRACT	7
H m HOLD	7
I t INPUT 6-BIT	7
-I t INPUT 4-Bit	7
I t SHIFT 6-Bits	8
-I t SHIFT 4-Bits	8
M m MULTIPLY	8
N m MULTIPLY	9
P t PRINT 6-Bit	9
-P t PRINT 4-Bit	9
R m SET RETURN ADDRESS	9
S m SUBTRACT	10
T m CONDITIONAL TRANSFER	10
-T m TRANSFER CONTROL	10
U m UNCONDITIONAL TRANSFER	10
Y m STORE ADDRESS	11

	PAGE
Zt STOP	11
Zt SENSE BS AND TRANSFER	11
-Zt SENSE OVERFLOW AND TRANSFER	11
 TIMING AND OPTIMIZING	 11
Sector Reference Timing Track	11
Timing	15
Optimization	15
Input/Output Timing	17
 COMPUTER CONTROL PANEL	 17
 INPUT/OUTPUT UNIT	 20
 OPTIONAL EQUIPMENT	 23
Model 101 Visual Oscilloscope Display	23
Model 141 Paper-Tape Reader	25
Model 151 Paper-Tape Punch	25

PROGRAM AND DATA MANIPULATION

	PAGE
 PROGRAM INPUT ROUTINE	 27
Bootstrap	27
Bootstrap Procedure	27
 SUBROUTINES	 28
Calling Sequence	28
 TAPE CODES	 28
 DATA INPUT	 29
 DATA OUTPUT	 29
 SCALING	 30
The q in Arithmetic Operations	30
 SHIFTING	 31
 NEGATIVE NUMBERS	 31
 HEXADECIMAL NOTATION	 31
 USE OF HEXADECIMAL ADDRESS	 32, 32A

TABLE I

PAGE

**DECIMAL AND HEXADECIMAL EQUIVALENTS OF COMMANDS
AND ADDRESSES**

32 B

TABLE II**POWERS OF 2 TABLE**

33

TABLE III

PAGE

INPUT/OUTPUT CODES

TABLES IV AND V	SHIFTING ACCUMULATOR TO RIGHT AND LEFT	3 ^{1/2}
HEXADECIMAL OPTIMIZATION CHART		36, 37

PAGE

ILLUSTRATIONS

FIGURE 1 — LGP-21 Computer System	1
FIGURE 2 — Word Structure	2
FIGURE 3 — Memory Disc	3
FIGURE 4 — Control Registers	4
FIGURE 5 — Sector Reference Timing Track	12
FIGURE 6 — Instruction Cycle - Phase 1	13
FIGURE 7 — Instruction Cycle - Phase 2	13
FIGURE 8 — Instruction Cycle - Phase 3	14
FIGURE 9 — Instruction Cycle - Phase 4	14
FIGURE 10 — Optimum Address Locator	15
FIGURE 11 — Optimum Operand Addresses	16
FIGURE 12 — Optimum Timing	17
FIGURE 13 — Computer Control Panel	18
FIGURE 14 — Model 121 Tape Typewriter	20
FIGURE 15 — Oscilloscope	24
FIGURE 16 — Oscilloscope Controls	24
FIGURE 17 — Model 141 Paper-Tape Reader	25
FIGURE 18 — Model 151 Paper-Tape Punch	26
FIGURE 19 — Tape Channels	28
FIGURE 20 — Input/Output Assignments	30
FIGURE 21 — 200.625 @ q = 8	30
FIGURE 22 — Extended Accumulator	30

LGP-21 COMPUTER SYSTEM

GENERAL CHARACTERISTICS

The LGP-21 is a compact digital computer with a large, 4096-word memory. It is completely mobile, can be plugged into any convenient outlet, and is fully transistorized for reliable operation and ease of maintenance.

This computer was designed as an economical solution for the general purpose computation needs of business, engineering and science. Operation and programming can be learned within a few days, even by personnel with no previous knowledge of computers. A compact vocabulary of 23 instructions is easily mastered and allows simple programming of complex problems.

The basic LGP-21 System includes the Model 21 Computer and the Model 121 Tape Typewriter input/output unit. For users who require greater input/output speeds, the following auxiliary equipment is also available: the Model 141 Paper-Tape Reader and the Model 151 Paper-Tape Punch, which process information at 60 characters-per-second.

Another auxiliary device for the LGP-21 is the Model 101 Visual Oscilloscope Display which provides a visual representation of the three LGP-21 control registers. This is useful in locating errors during program checkout, and in providing a visual indication of the key locations at which a program may stop during operation.

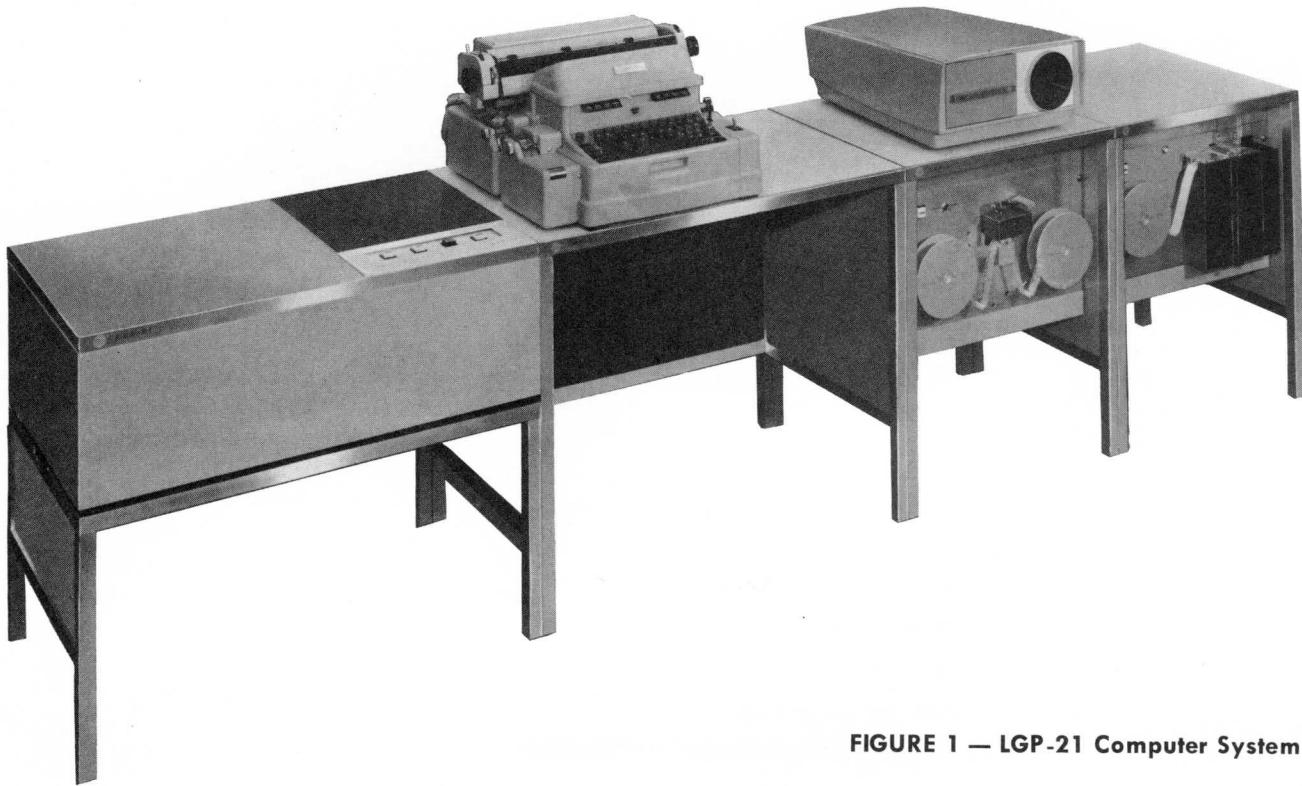


FIGURE 1 — LGP-21 Computer System

Input/Output Device

The LGP-21 System will accommodate up to 32 input and output devices in any desired combination. They are selected individually under program control. Output operations can be performed on more than one device.

The primary input/output device for the LGP-21 is the Model 121 Tape Type-writer. This device has a standard alphanumeric keyboard, a paper-tape reader, a paper-tape punch, and special control levers.

Information can be input automatically from the tape reader to the typewriter and then to the computer at 10 characters-per-second, or manually from the typewriter keyboard to the computer. Output information from the LGP-21 can be recorded by the typewriter or by the typewriter and tape punch at 10 characters-per-second. Input and output data is normally in decimal notation, while internal computation is performed in binary by the LGP-21. Conversion to binary, for this purpose, is achieved automatically through programming.

Word Structure

The LGP-21 processes all information as 32-bit words ("bit" is derived from "binary digit"). Each word consists of a sign bit, 30 bits of information, and a spacer bit. When dealing with binary values, the presence of a bit value is indicated by "1", the absence by "0". The sign bit contains a 1 to represent a negative word, or a 0 to represent a positive word. The spacer bit is used to separate adjacent words on the memory disc and is always recorded as 0.

The binary configuration of the 32 bits may represent either a numerical value (data word) or a computer instruction (instruction word). Data and instructions may be intermixed in memory, and the same unit of information can be used as data or instruction at different points in a program.

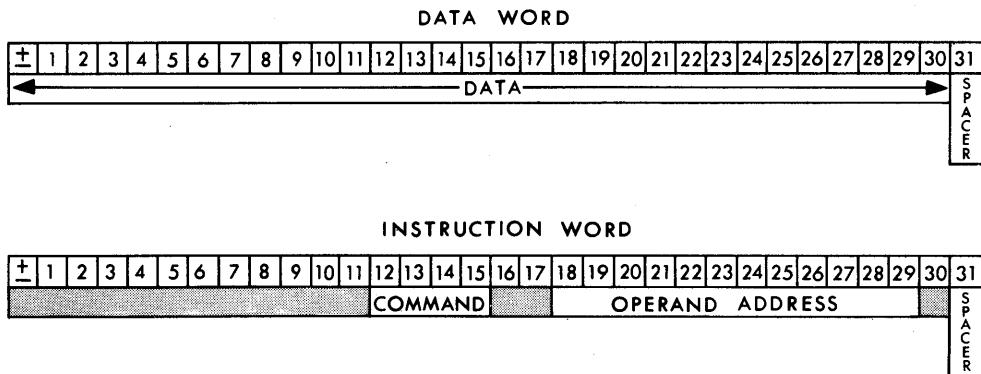


FIGURE 2 — Word Structure

When a word is used for data, the numbers are normally held in binary notation. The 30 significant binary digits in an LGP-21 word can represent decimal values up to 1,073,741,823. Decimal numbers can also be represented by a system of notation called binary-coded decimal. As many as eight decimal digits, representing any positive number less than 80,000,000, can be held in a binary-coded decimal word. Various capabilities for negative numbers and fractions can be deduced. In place of a numerical value, a word may contain up to five alphanumeric characters. Two or more binary or decimal numbers can be packed into the same word as long as their combined requirements do not exceed 30 bits plus sign.

When a word is used to represent an instruction, it contains one basic command (bits 12 through 15) and an operand address (bits 18 through 29). The remaining bits are available—except for the sign bit which is significant in the case of

certain commands—and do not affect the instruction. The command determines what type of operation is to be performed, i.e., addition, subtraction, division, etc. The operand address for most instructions identifies the location within memory of the quantities related to the operation. For transfer instructions it will contain the address of the next instruction to be executed in the event of an active transfer.

MEMORY

Information is recorded in the form of magnetized spots on the surface of a rotating disc. When the disc is idle, the read/write heads rest on its surface; but when the disc attains operating speed, air pressure causes the heads to float at the correct operating position. The memory disc revolves at 1180 rpm. Thus, one revolution requires about 52 milliseconds. One word-time—the time required for one word to pass under a read head—is therefore about 0.40 millisecond.

Figure 3 illustrates the relationship of the sectors to their respective read/write heads.

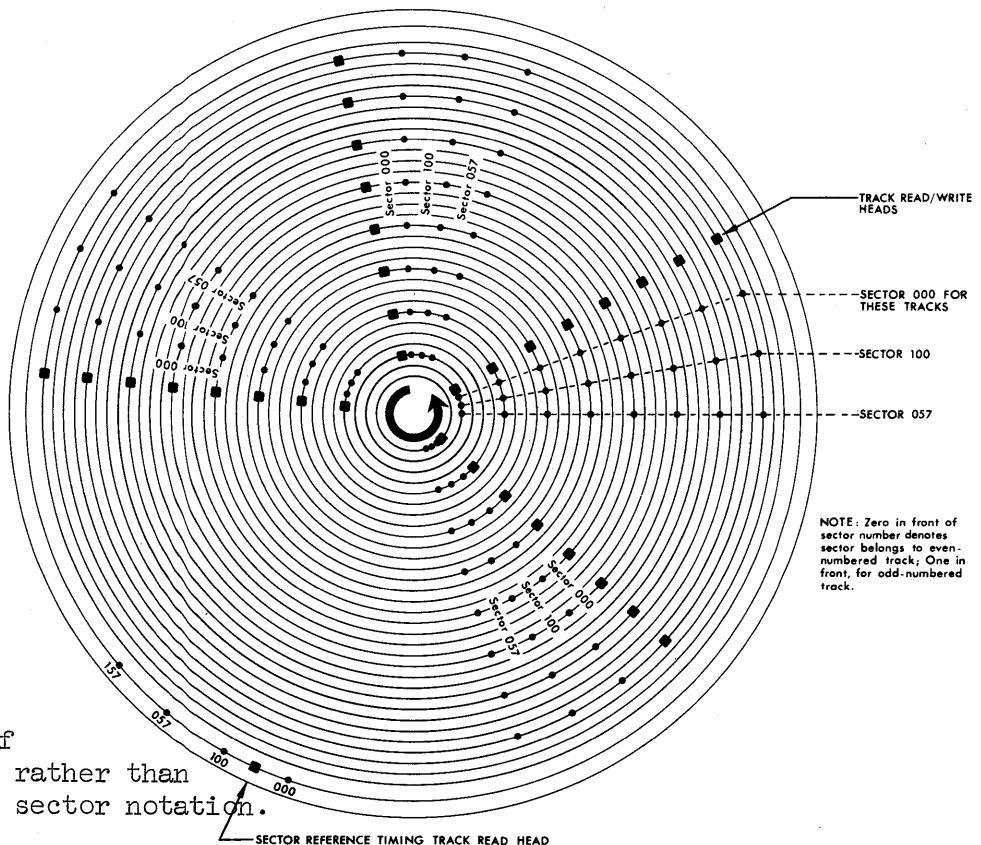


FIGURE 3 — Memory Disc

The memory disc has a total capacity of 4096 words or sectors. The sectors are divided into logical groups of 64 called tracks. There are a total of 64 tracks in memory. Both tracks and sectors are numbered 00 through 63. The location of any word in memory can be specified by its track and sector number. This number is known as the address of the word. For example, 2347 is the address of the word in Track 23, Sector 47. Consecutive addresses are 0000, 0001, 0002... 0063, 0100... 6361, 6362, 6363, 0000.

Mounted above the surface of the memory disc are 32 read/write heads which inscribe concentric circles as the disc revolves. Each read/write head serves two tracks, which are assigned alternate sectors in a circle. Thus read/write head number 1 reads the first sector of track 00, then the first sector of track 01, second sector of track 00, second sector of 01, and so on.

The sectors are not numbered sequentially within the tracks, but the pattern of the numbering is the same for all. It is based on an 18-word interlace which positions consecutive words 18 sectors apart and provides a 7.26 ms timing interval between them. This permits the optimizing of operand addresses for instructions which are executed in sequence.

COMPUTER CONTROL

All computer functions are controlled by three registers and their associated circuitry which comprise the arithmetic and computing control elements. The three control registers are designated: Accumulator (A), Instruction Register (R), and Counter Register (C). These registers are recirculating lines in a single track on the memory disc. A recirculating line has a read-head and a write-head, located 1 word-time apart, which continually copies information from one sector into another. Thus, the same word is available at every word-time.

A fourth register on the same track is the Extended Accumulator. Its read-head and write-head are located 2 word-times apart. This register records the double-length product of the multiply operation. Depending upon the command used, the most or least significant half of the double-length product is recorded in the normal Accumulator. This half can be stored in memory for future use.

Figure 4 shows the relationship of the registers on the recirculating line.

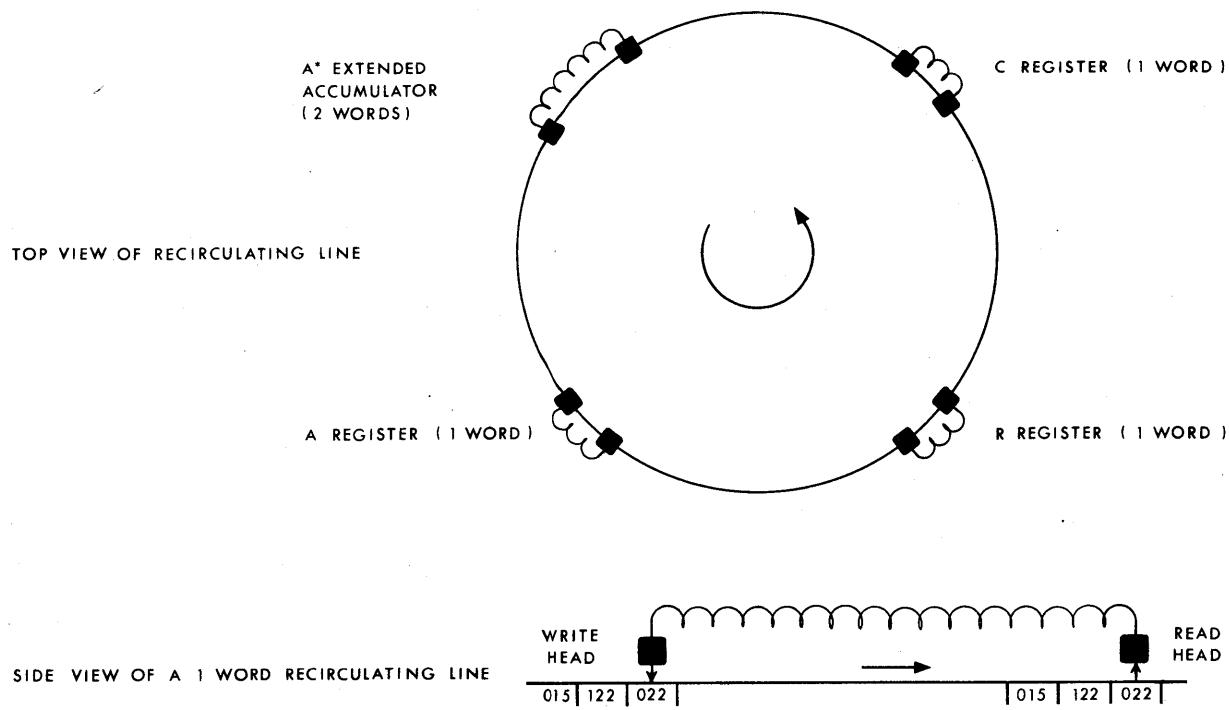


FIGURE 4 — Control Registers

Accumulator

The Accumulator, register A, is the working register used for all manipulation of data. It is the register through which all information passes between the computer and its input/output devices, as well as from one part of memory to another.

The Accumulator contains one of the operands prior to execution of an arithmetic instruction and the result of the arithmetic operation after execution. The second operand is a word in memory whose location is specified by the address portion of the instruction. The structure of the Accumulator is the same as that of a memory word (a sign and 31 bits), except that the spacer bit in the Accumulator can contain a 0 or 1 as the result of input. A multiply operation develops a double-length product (62 bits), either half of which can be retained in the Accumulator.

Instruction Register

The Instruction Register, R, holds the instruction being executed by the computer. Instructions in the stored program are transferred one at a time in sequence to the Instruction Register for execution. The contents of this register specifies the operation to be performed and the word in memory with which or upon which to operate. The configuration of the Instruction Register is the same as that of an instruction word.

During multiply and divide operations, the Instruction Register contains the multiplier or divisor rather than the instruction.

Counter Register

The Counter Register, C, contains the address of the next instruction to be executed. After executing an instruction, the computer searches for the word whose address is in the Counter. That word is transferred to the Instruction Register to be executed, and the contents of the Counter are incremented by 1. There are two commands (T and U) which set the Counter Register to a specified address.

If overflow occurs—that is, if a quantity beyond the capacity of the Accumulator has been generated—due to the execution of an add, subtract, or divide instruction, a 1 is recorded in the sign position of the Counter Register. The overflow bit may be examined and reset to 0 by a negative Sense and Transfer instruction, for example, 800Z0200.

COMMAND LIST

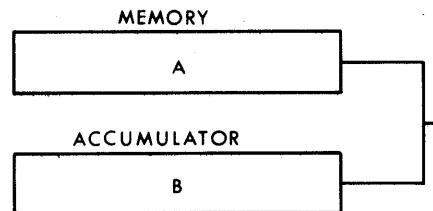
The Command List

The LGP-21 has a repertoire of 23 commands. An instruction is a combination of a command and an operand address. Each instruction commands the computer to perform one of its built-in operations. The following list shows the commands and the effect of each. When the address is shown as "m", any memory location could be designated as the operand of that instruction. When the address is shown as "t", the track number is significant as a code, while the sector number is irrelevant and does not affect any function of the instruction.

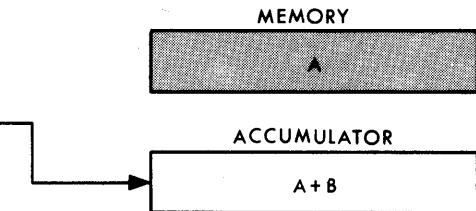
A m ADD

Add the contents of location m to the contents of the Accumulator; the sum replaces the contents of the Accumulator. If an addition results in a number that is outside the limits of the Accumulator, overflow will occur. Memory remains unaltered.

INITIAL CONTENTS

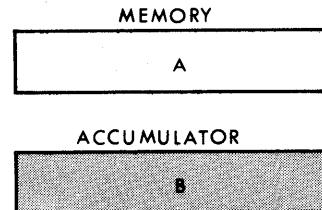


FINAL CONTENTS

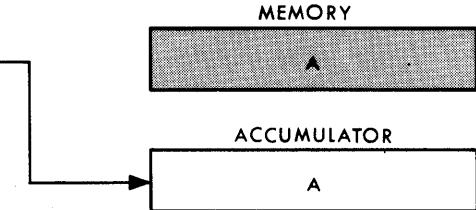
**B_m BRING**

Bring the contents of location m to the Accumulator, replacing its contents. Memory remains unchanged.

INITIAL CONTENTS

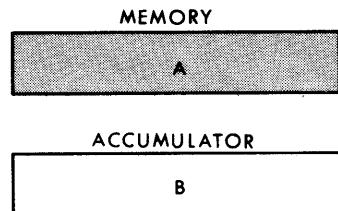


FINAL CONTENTS

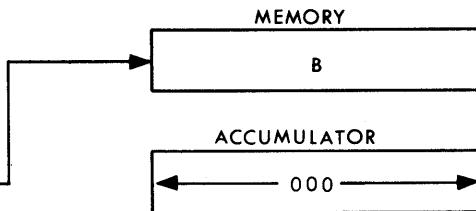
**C_m CLEAR**

Store the contents of the Accumulator in memory location m; then clear the Accumulator to zero.

INITIAL CONTENTS

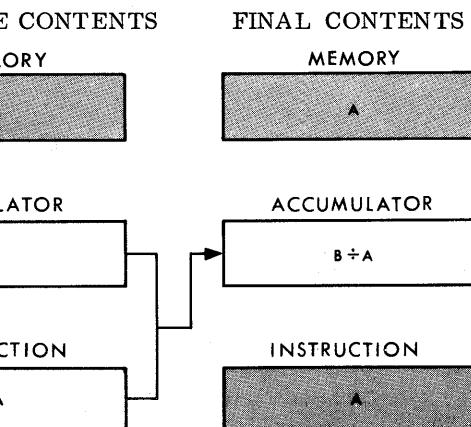
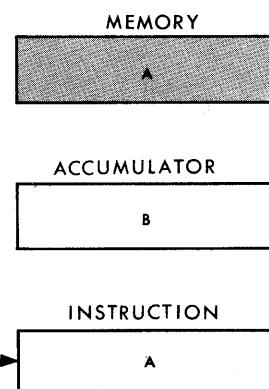
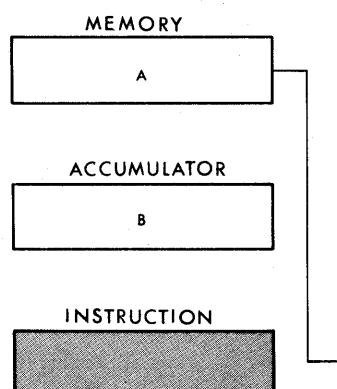


FINAL CONTENTS

**D_m DIVIDE**

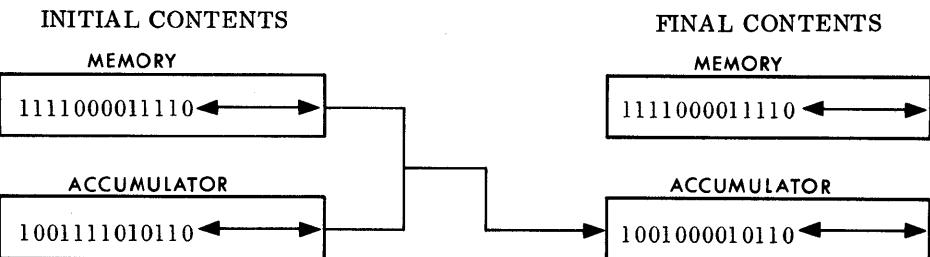
Divide the number in the Accumulator by the number in location m, retaining the quotient, rounded to 30 bits, in the Accumulator. The absolute value of the contents of m must be greater than the absolute value of the contents of the Accumulator, or overflow will occur. During the divide operation, the Instruction Register holds the divisor.

INITIAL CONTENTS INTERMEDIATE CONTENTS FINAL CONTENTS



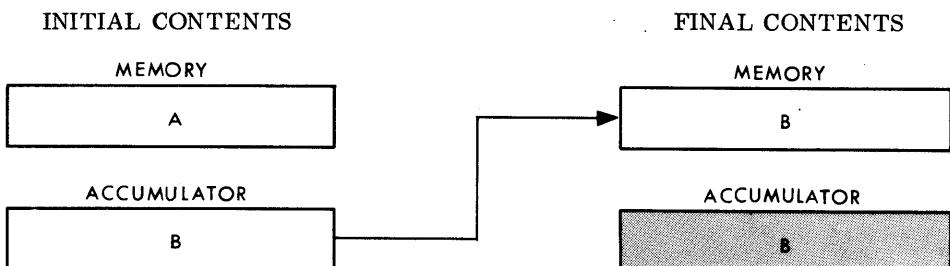
E m EXTRACT

Where "1" bits are in location m, retain the value of the corresponding bit positions in the Accumulator; where "0" bits are in m, place 0 bits in the corresponding positions in the Accumulator. The word in m is called the "mask" and remains unaltered.



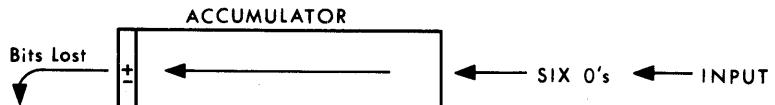
H m HOLD

Store the contents of the Accumulator in location m, without altering the contents of the Accumulator.



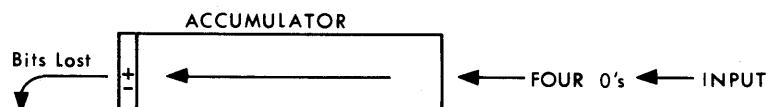
I t INPUT 6-BIT

Shift the contents of the Accumulator left 6 bits, inserting zeros at the right; then give a start read signal, allowing the 6 bits of each character read by the input device specified by t to enter the Accumulator. (See Input Note below.)



-I t INPUT 4-Bit

Shift the contents of the Accumulator left 4 bits, inserting zeros at the right; then give a start read signal, allowing 4 bits of each character read by the input device specified by t to enter the Accumulator. (See Input Note below.)



Input Note: Input is terminated when a stop code is read following the least significant character. The Model 141 Paper-Tape Reader can also enter fixed word-length items by any method of programming that leaves the binary configuration of a stop code in a particular position in the Accumulator to delimit the word length. When the 4-bit (1000) or 6-bit (100000) configuration of a stop code is pushed past bit position zero of the Accumulator, input will be stopped.

The fixed word-length mode of input cannot be used with the typewriter since it needs to read a stop code to halt its reader. The 141 Paper-Tape Reader will

cease input when a stop code is read or when a stop code is pushed out the Accumulator on the left, whichever occurs first.

The following table shows hexadecimal constants that will position a stop code to delimit the word length.

<u>Number of Characters</u>	<u>4-Bit Input</u>	<u>6-Bit Input</u>
1	08000000	02000000
2	00800000	00080000
3	00080000	00002000
4	00008000	00000080
5	00000800	00000002
6	00000080	-----
7	00000008	-----

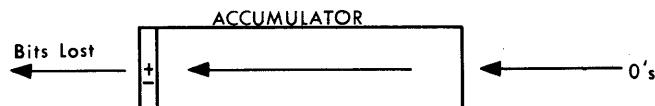
I t SHIFT 6-Bits

When $t = 62$, shift the contents of the Accumulator left 6 bits, inserting zeros at the right.



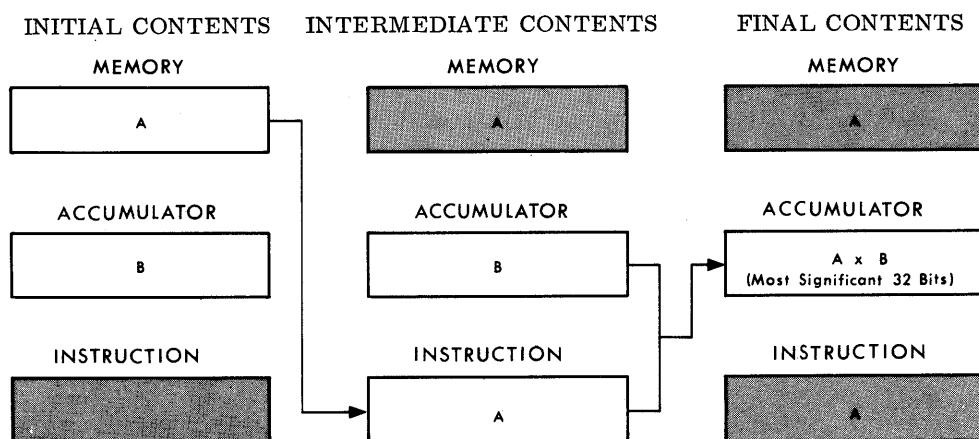
-I t SHIFT 4-Bits

When $t = 62$, shift the contents of the Accumulator left 4 bits, inserting zeros at the right.



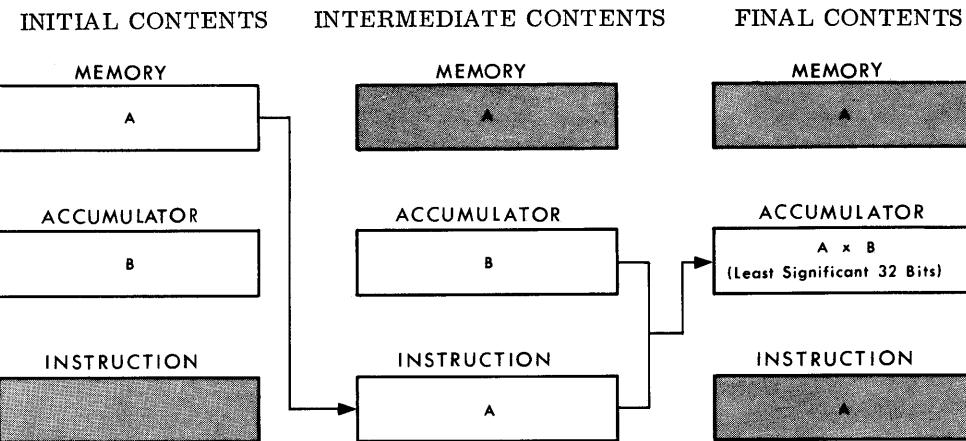
M m MULTIPLY

Multiply the contents of the Accumulator by the contents of location m, forming a 62-bit product of which 32 bits are retained: the sign and the most significant 31 bits of the product replace the contents of the Accumulator. The Instruction Register holds the multiplier during the multiply operation.



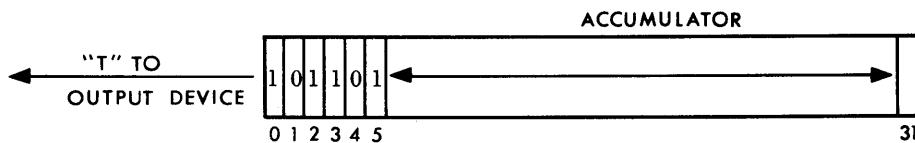
N m MULTIPLY

Multiply the contents of the Accumulator by the contents of location m, forming a 62-bit product of which 32 bits are retained: the least significant 32 bits replace the contents of the Accumulator, occupying bit positions 0 through 31. Loss of any of the most significant bits does not cause overflow. This instruction is used mainly to accomplish left shifts. During the multiply operation, the Instruction Register holds the multiplier.



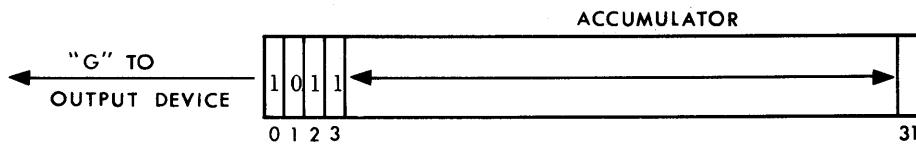
P t PRINT 6-Bit

Transmit the character represented by bits 0 through 5 of the Accumulator to the output device specified by t. The contents of the Accumulator remain unaltered. (See Table III, Input and Output Codes.)



-P t PRINT 4-Bit

Combine "1" for channel 5 and "0" for channel 6 with bits 0 through 3 from the Accumulator, then transfer this hexadecimal character to the output device specified by t. The contents of the Accumulator remains unaltered. (See Table III, Input and Output Codes.)



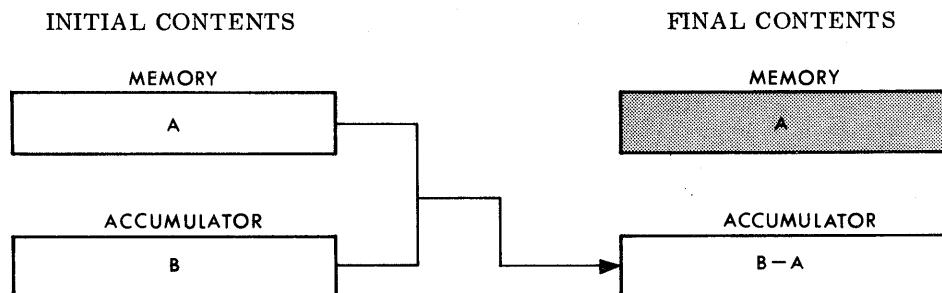
R m SET RETURN ADDRESS

In the address portion of location m, record the address which is 2 greater than the location of the R instruction being executed (i.e., the contents of the Counter Register plus 1). The rest of the memory word remains unaltered.



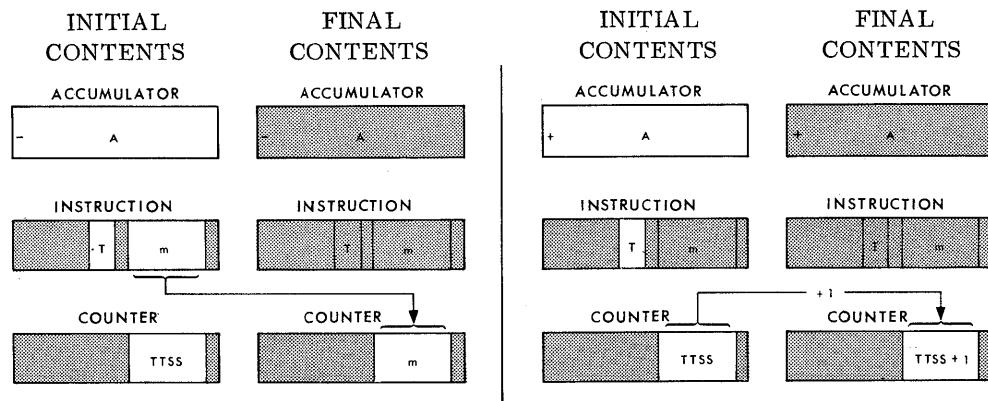
S_m SUBTRACT

Subtract the contents of location m from the contents of the Accumulator and retain the difference in the Accumulator. If a subtraction results in a number that is outside the limits of the Accumulator, overflow will occur. Memory remains unaltered.



T_m CONDITIONAL TRANSFER

If the contents of the Accumulator is negative (1 in the sign position), replace the contents of the address portion of the Counter Register with m and get the next instruction from location m. Otherwise, continue to the next instruction in sequence without altering the Counter.

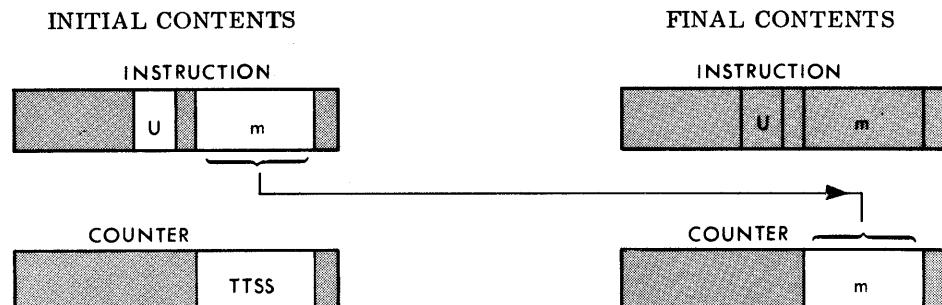


-T_m TRANSFER CONTROL

If the contents of the Accumulator is negative or if the TC switch on the console is ON, replace the contents of the address portion of the Counter Register with m and get the next instruction from location m. Otherwise, continue to the next instruction in sequence without altering the Counter.

U_m UNCONDITIONAL TRANSFER

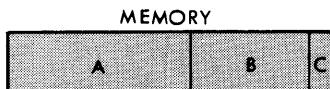
Replace the contents of the address portion of the Counter Register with m and get the next instruction from location m.



Y_m STORE ADDRESS

Replace the address portion of the word in location m with the address portion of the word in the Accumulator, leaving the rest of m and all of the Accumulator undisturbed.

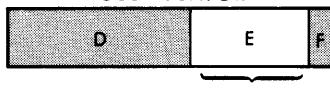
INITIAL CONTENTS



FINAL CONTENTS



ACCUMULATOR



Z_t STOP

Halt computations when t = 00 or 01.

Z_t SENSE BS AND TRANSFER

Interrogate the Branch Switch specified by t. If the switch is OFF, skip the next instruction in sequence. If the switch is ON, execute the next instruction in sequence. The Branch Switches are numbered 4, 8, 16, and 32. (See Sense Note below.)

-Z_t SENSE OVERFLOW AND TRANSFER

If overflow is off (0 in the sign position of the Counter Register), skip the next instruction in sequence. If overflow is on (1 in the sign position of the Counter), reset the overflow bit to zero; then execute the next instruction. The track number (t) designates which, if any, Branch Switches are also to be interrogated. If Sense Overflow is combined with Stop (800Z0000), the skip or no skip is deferred until after the stop. If no Branch Switches are to be tested and no stop is desired, the track can be 02 or 03. (See Sense Note below.)

Sense Note: Overflow and/or any combination of Branch Switches can be interrogated with one Sense and Transfer instruction. If any are OFF, the next instruction will be skipped. If all are ON, the next instruction will be executed.

TIMING AND OPTIMIZING

Sector Reference Timing Track

In the LGP-21 all instructions are executed in sequence. The time required for completing a specific operation depends, in part, on the location in memory of the instruction and of its operand, if one is necessary. The process by which the computer obtains and executes an instruction is called an instruction cycle. An instruction cycle begins with a memory search for the instruction word and ends with the commencement of the search for the next instruction word.

The complete cycle consists of four phases:

Phase 1 - Search for the instruction.

Phase 2 - Transfer the instruction from main memory to the Instruction Register and increment the Counter Register by 1.

Phase 3 - Search for the operand.
Phase 4 - Execute the instruction.

In order for the computer to find a specific location in memory, a Sector Reference Timing Track is used. This track contains the sector numbers 00 through 127 permanently pre-recorded at the time of manufacture. As explained under "MEMORY," each of the concentric circles on the disc contains 128 sectors. Sector addresses are numbered 00 through 63 for programming purposes. Therefore, on the Sector Reference Timing Track numbers greater than 63 are interpreted as modulo 64. For example, sector 97 is sector 33 (i. e., $97 - 64 = 33$) for all odd-numbered tracks.

The Sector Reference Timing Track has only a read head and cannot be modified by the programmer. The numbers on this track pass under its read head one sector before the corresponding sector in main memory passes under its read head. Thus, when a specified sector address is read on the Sector Reference Timing Track, the read/write head on the appropriate track is activated, and the word can be read from or recorded in memory. For example, assume the contents of Location 1432 is to be brought to the Accumulator. When the Sector Reference Timing Track reads 32, read head 14 is activated, and as sector 32 moves under that read head, its contents is copied into the Accumulator. See Figure 5.

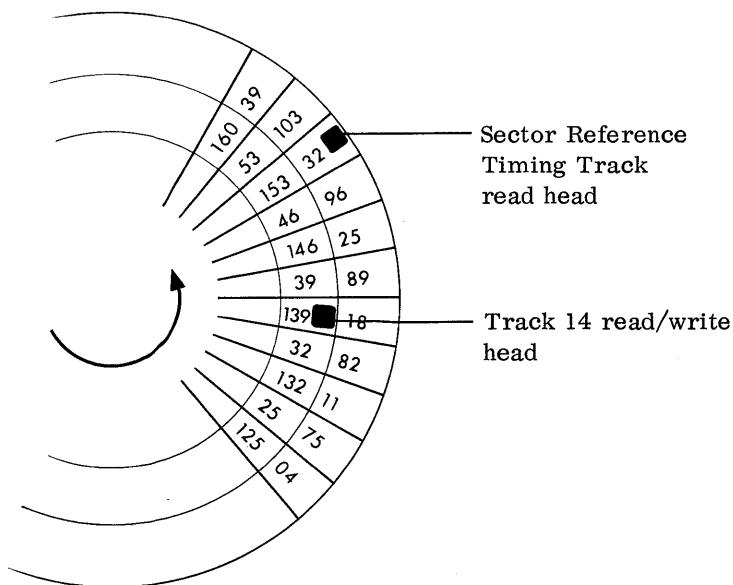


FIGURE 5 — Sector Reference Timing Track

This may be more easily understood if two instructions are considered in sequence. Assume that the instruction B4458 is stored in memory Location 1115, and the instruction A4452 is in Location 1116. When executed, these instructions will result in some value—say 8—being brought to the Accumulator from 4458 and another value—say 7—being added to it.

During Phase 1 the Counter Register contains the address 1115, so the computer searches the Sector Reference Timing Track for sector 79 ($79 - 64 = 15$). When it is read, the read head on track 11 is activated, and Phase 1 ends. (Figure 6)

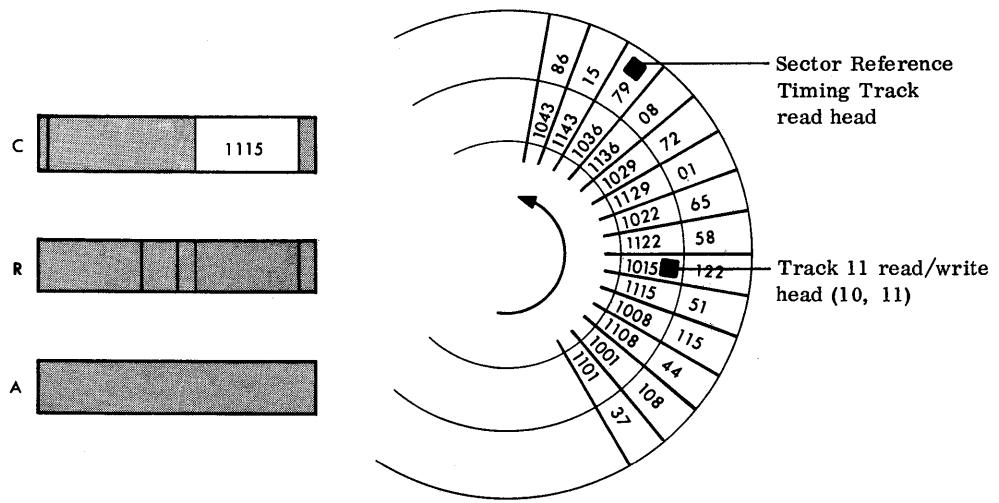


FIGURE 6 — Instruction Cycle - Phase 1

In Phase 2 the contents of Location 1115 is read into the Instruction Register, and the Counter Register is incremented by 1, so that it now contains 1116. (Figure 7).

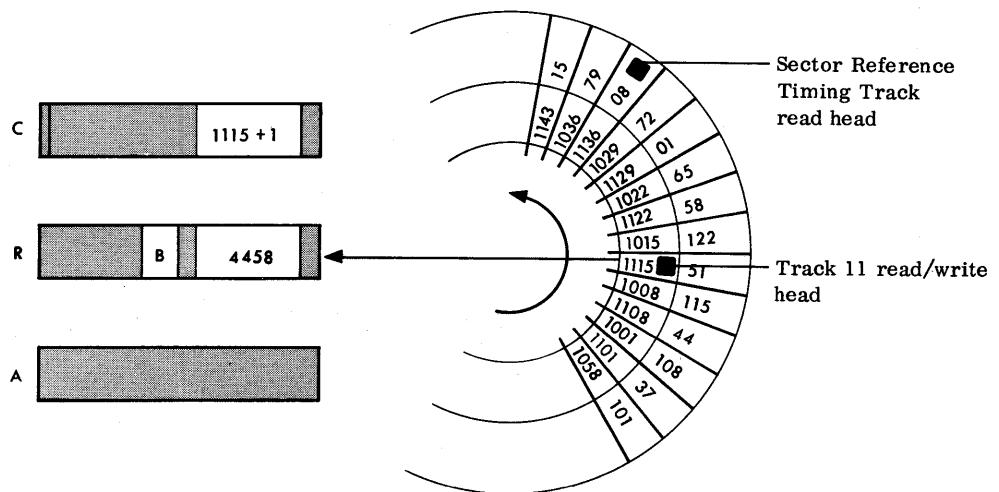


FIGURE 7 — Instruction Cycle - Phase 2

During Phase 3 the computer searches the Sector Reference Timing Track for the sector specified in the Instruction Register—that is, sector 58. (Figure 8).

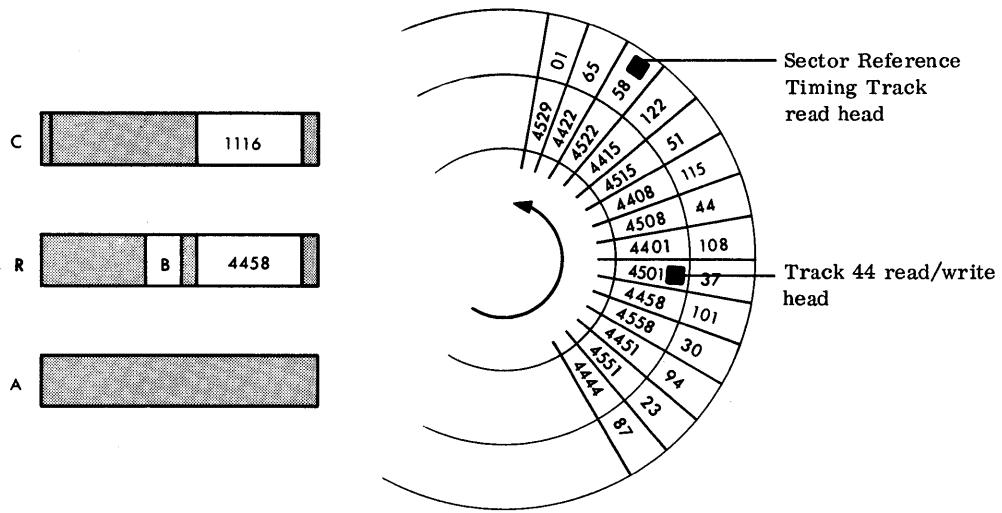


FIGURE 8 — Instruction Cycle - Phase 3

When sector 58 is read, Phase 3 ends, and the computer goes to Phase 4 to execute the instruction B4458. Therefore, the contents of Location 4458 (the number 8) is copied into the Accumulator. (Figure 9).

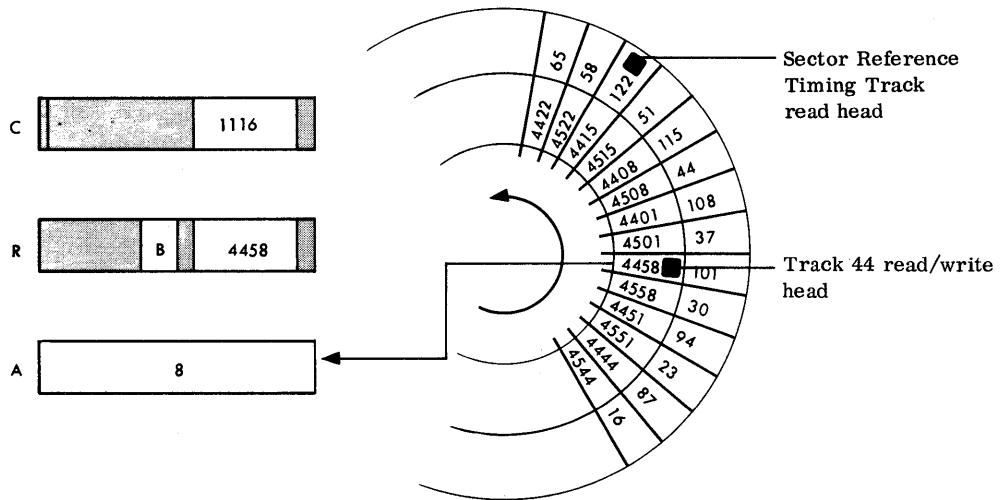


FIGURE 9 — Instruction Cycle - Phase 4

Then the cycle begins again:

<u>Phase</u>	<u>Activity</u>
--------------	-----------------

- 1 Counter Register contains 1116, so search for sector 80 on the Sector Reference Timing Track. When sector 80 is found, activate the read head for Track 11.
- 2 Copy contents of Location 1116 (A4452) into the Instruction Register. Increment Counter Register to 1117.

- 3 Search for sector 52. When it is read, activate the read head for Track 44.

- 4 Execute the instruction; that is, add the contents of 4452 (the number 7) to the contents of the Accumulator (8) and leave the sum (15) in the Accumulator.

Note: When a Sense and Transfer instruction results in a "skipped" instruction, the instruction is actually brought into the Instruction Register, but is not executed. That is, Phases 3 and 4 of the skipped instruction are bypassed, and the computer advances directly to Phase 1 of the following instruction.

Timing

Generally, if all four phases of an instruction can be executed before the disc turns past the location of the next instruction in sequence, the instruction is said to be optimum. However, there are a few instructions which require more time for their operations than the four phases allow. These exceptions include the multiply, divide, input, and output operations.

During Phases 1 and 3 the computer searches for the specified sector and then activates the appropriate read/write head. Either of these phases can require one or more word-times. Phase 2 is always one word-time. Phase 4 requires one word-time for all instructions except M Multiply, N Multiply, and Divide. These three (N, M, and D) require 63, 65, and 66 word-times respectively.

Optimization

Figure 10 shows a device which is useful in determining the optimum sector for the operand of any instruction. For example, if an Add instruction is stored in

See page 32 for hexadecimal optimum address calculations.

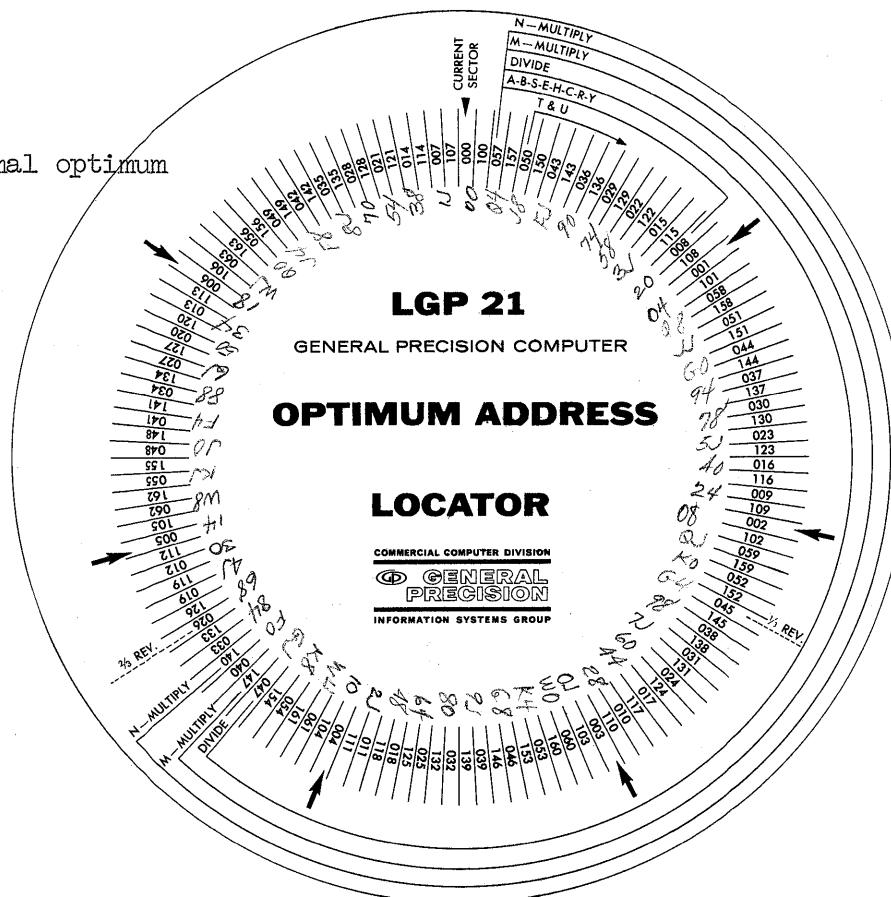


FIGURE 10 — Optimum Address Locator

sector 00 of a track, it is optimum when it can be executed before sector 01 passes under the read head. By referring to Figure 10 it can be seen that for an optimum Add instruction in sector 00 of an even-numbered track, its operand must be in one of the following sectors: 57, 50, 43, 36, 29, 22, or 15 on an odd-numbered track (indicated on the wheel by the initial digit "1"), or 57, 50, 43, 36, 29, 22, 15, or 08 on an even-numbered track (indicated by the initial digit "0").

Figure 11 shows the LGP-21 commands and the optimum operand addresses for an instruction in an even-numbered track, sector 00.

Instruction in Location 000	Optimum Operand Addresses	Distance from Instruction location to Optimum Operand (in word-times)
Bring, Add, Subtract, Hold, Clear, Extract, Set Return Address, Store Address	057, 157, 050, 150, 043, 143, 036, 136, 029, 129, 022, 122, 015, 115, 008	2 through 16
N-Multiply	057, 157, 050, 150, 043, 040, 140	2 through 81
M-Multiply	057, 157, 050, 150, 043, 047, 147	2 through 79
D-Divide	057, 157, 050, 150, 043, 154, 047	2 through 78
Unconditional Transfer, Conditional Transfer	050, etc.	4 or more
Others	unrelated	unrelated

FIGURE 11 — Optimum Operand Addresses

Depending on whether an instruction is optimized or not, certain time intervals are required for the computer to read an instruction, execute it, and be ready to read the next instruction. Figure 12 shows the varying timing requirements:

Instruction	Time in Milliseconds	
	Optimum	Non-optimum
Bring, Add, Subtract, Hold, Clear, Extract, Set Return Address, Store Address, Input Shift	7.26	58.11
N-Multiply, M-Multiply, D-Divide	58.11	108.96
Unconditional Transfer, Conditional Transfer	1.60	Each sector beyond optimum adds .40
Sense	7.26 or 14.52	

FIGURE 12 — Optimum Timing

Input/Output Timing

An input instruction will be delayed within its cycle until a start signal is received from the input device. Three word-times plus the Phase 1 search, plus the time required for reading or typing determine the total time for any input instruction.

Print instructions should be timed at least 2 revolutions apart for the 121 Tape Typewriter or 1/3 revolution apart for the 151 Tape Punch if they are to operate at their rated speeds. An output operation will not delay the computer unless the selected device is already in use. For example, if an output to the tape typewriter is followed by an output to the 151 Tape Punch, the computer will not delay on the second output instruction even though the tape typewriter is still busy. If the first and second output instructions both select the tape typewriter, the computer will delay on the second instruction until the typewriter is ready.

COMPUTER CONTROL PANEL

Operational control of the computer is effected through the control panel switches, all of which are clearly identified by function or related action. See Figure 13.

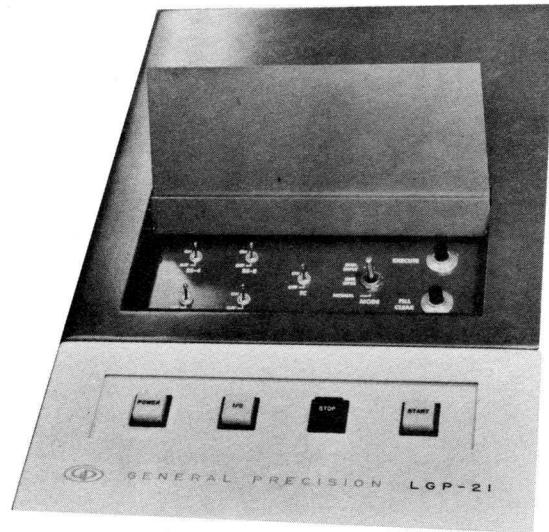


FIGURE 13 — Computer Control Panel

MODE

This is a three-position toggle switch:

NORMAL - When the MODE switch is set to this position, the computer executes instructions at high speed. When the START switch is depressed, the execution of instructions will begin, commencing with the instruction whose address is contained in the Counter.

ONE OPERATION - If the computer is operating in Normal mode and the MODE switch is moved to ONE OPERATION, the computer will stop after the next Phase 4, the execute phase. If the computer is stopped in One Operation mode, depressing the START switch will start the computer cycling through the instruction whose address is in the Counter Register, and computation will stop after the execute phase for that instruction. The EXECUTE switch is operative only in One Operation mode. Going from Manual Input mode to One Operation mode will de-select the tape typewriter.

MANUAL INPUT - This position sets the Accumulator to receive input. It also selects the typewriter for 4-bit input, but does not de-select other devices. If other devices are selected, the I/O switch should be depressed, or information may not enter correctly. All typed characters enter the Accumulator. No instruction can be executed in Manual Input mode since the START switch is inoperative.

FILL CLEAR

FILL CLEAR is a momentary switch which transfers the contents of the Accumulator to the Instruction Register and resets the Counter Register to zero. This switch is inoperative in Normal mode.

EXECUTE

This momentary switch causes the instruction in the Instruction Register to be executed. It is operative only in One Operation mode.

Transfer Control

The TC switch can be set ON or OFF. This switch is used in conjunction with the negative T (Conditional Transfer) command. A negative T instruction will cause the computer to get the next instruction from the location designated by the operand address if the contents of the Accumulator tests negative or if the TC switch is ON, otherwise it causes the computer to continue to the next instruction in sequence.

Branch Switches

The four branch switches are labeled BS-32, BS-16, BS-8, and BS-4. Each is a two-position switch which is set ON or OFF. These switches are used in conjunction with the Z (Sense and Transfer) command. A Z instruction whose track-address corresponds to one or more of the branch switches will cause the computer to skip the next instruction if any designated switches are OFF, or to execute the next instruction if all designated switches are ON.

POWER

This switch turns power ON or OFF. Power for all units in the system is in series with this switch. Any units that are previously set ON will have their power turned ON as the switch is depressed. About thirty seconds after power is turned ON, the switch will light to indicate the machine has attained full power.

I/O

I/O is a momentary switch which clears the Accumulator and de-selects all input and output devices. If the computer is in Manual Input mode, I/O will not de-select the typewriter. The switch is lighted and operative during input and output and when the computer is in Manual Input mode.

STOP

This indicator lamp lights immediately when the computer is turned ON, and is lit whenever the computer is not executing instructions.

START

START is a momentary switch which causes the computer to execute the instruction specified by the Counter Register. In Normal mode this will begin the full-speed execution of instructions. In One Operation mode, only one instruction will be executed. The switch is not operative in Manual Input mode. The light beneath the switch is ON whenever the computer is operating.

In addition to those on the control panel, there are two toggle switches on back of the computer:

INTERLOCK

The LGP-21 has a circuit breaker to interrupt operation if the air-flow from the fan becomes blocked. This interruption stops computer operation to prevent overheating. Following such an interruption, the condition that caused it should be corrected; then the circuit may be reset by moving the INTERLOCK switch from the up position down and up again. It should be noted that, depending upon the operation in effect when the interlock occurred, information stored in memory may have been destroyed and may have to be re-entered.

WRITE ENABLE

This switch may be set ON or OFF. When it is ON, reading from and recording in all sectors of all tracks may occur. When it is set to OFF, 1024 words—specifically Tracks 00 through 15—are protected; that is, information may be transferred from any word within this area to the Instruction Register or the Accumulator, but no information can be recorded in any word in this area. This

feature allows the operator to lock a program in memory so that it cannot be destroyed inadvertently.

INPUT/OUTPUT UNIT

The primary input/output unit for the LGP-21 is the Model 121 Tape Typewriter. In addition to a standard typewriter keyboard, the unit has a paper-tape punch, paper-tape reader, and various levers for controlling their operations. See Figure 14.

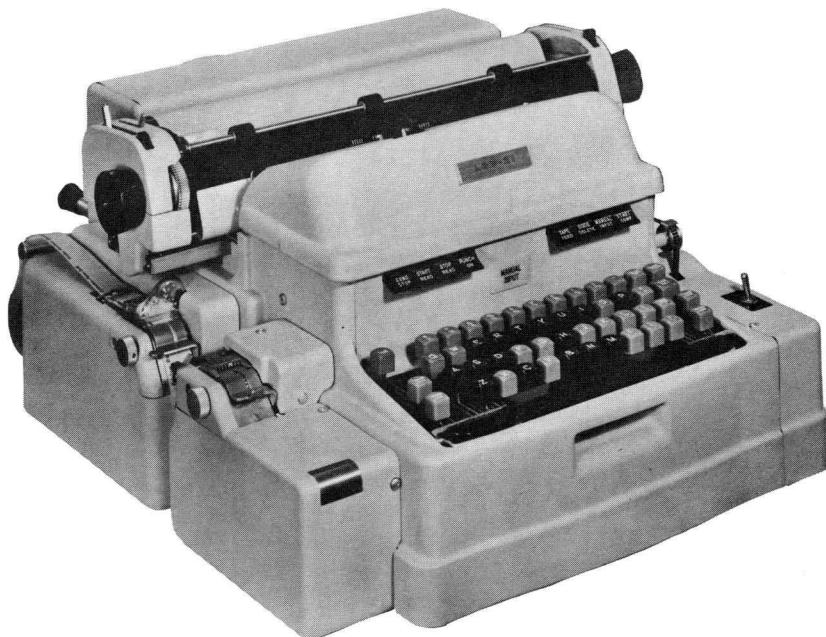


FIGURE 14 — Model 121 Tape Typewriter

POWER ON-OFF

This switch, in the lower right-hand corner adjacent to the keyboard, sets the typewriter so that power will be turned ON or OFF when the computer power is ON. The carriage is interlocked and should not be moved when power is OFF.

START COMP

When the computer has selected the typewriter for input, the START COMP lever terminates input and allows the computer to proceed to the next instruction. The START COMP lever will stop the paper-tape reader whenever it is running on-line or off-line. The functions of the STOP READ lever are identical to the START COMP lever.

MANUAL INPUT

If this lever is down and the typewriter is selected for input, information can be transmitted to the computer from the keyboard only. If this lever is raised, information is received from the tape reader when the typewriter is selected for input.

CODE DELETE

Operative only when the PUNCH ON lever is depressed, this lever is used to delete an error in the tape by punching holes in channels 1 through 6. The tape must be manually backspaced one character to delete the error. By holding down TAPE FEED and CODE DELETE at the same time, a series of delete codes can be punched.

TAPE FEED

This lever feeds tape into the punch, which then punches only sprocket feed holes. TAPE FEED is operative only when PUNCH ON is depressed.

Manual Input Light

This light is illuminated when the typewriter is selected for input to the computer and the MANUAL INPUT lever is down.

PUNCH ON

The PUNCH ON lever turns the tape punch ON, allowing any character typed from the keyboard or read from the tape reader to be punched. TAPE FEED and CODE DELETE are operative only when PUNCH ON is depressed. Raising the lever turns the punch OFF.

STOP READ

Used interchangeably with START COMP.

START READ

This lever starts the tape reader. Reading will continue until a Conditional Stop Code is read, providing the COND STOP lever is raised; or until the STOP READ or START COMP lever is depressed.

COND STOP

This lever, when depressed, allows the tape reader to read without being stopped by the Conditional Stop codes. This lever must be raised during input to the computer from the tape reader.

Paper Guide

Located just to the rear of the platen, this guide should be adjusted horizontally so that it just touches the left edge of the paper form.

Tab Stop

Under the cover to which the paper guide is attached is the tab rack numbered 8 through 136 in increments of 4. A tab stop is a metal positioner that can be inserted in any notch along the tab rack. When the TAB key is depressed or the Tab code is read, the carriage will move to the next position containing a tab stop.

Left Margin Stop

In front of the tab rack is the margin rack numbered 0 through 68 in increments of 4. The margin stop is the sliding assembly mounted on the margin rack. To move this assembly, press down on its center and slide it along the rack. The right end of the stop is the indicator. The setting of the margin stop determines the left margin position.

Automatic Carriage Return

Behind the tab rack is a carriage return plate. An automatic carriage return positioner can be placed anywhere along the plate. An automatic return occurs when the carriage reaches this return positioner as the result of a tab jump; i.e., because the Tab key is depressed, a Tab code is read from tape, or the computer outputs a Tab code. If this positioner is reached as a result of single-

character steps, the typewriter may jam. This condition may be cleared by striking the Carriage Return key manually. However, any input or output that occurred at the time of the jamming may be invalid.

Paper Scale

The paper scale is printed on the metal shield in front of the platen. By viewing the paper scale through the type guide, one can determine the exact position of the carriage and exactly where characters will print.

Type Guide

This guide indicates the position of the carriage and the location where the characters will print.

Writing Line Finder

The bottom of the typed line will be exactly above the top edge of the writing line finder. It is used to align a previously typed page in the platen for additional typing.

Paper Release

The paper release lever is located at the top left-hand corner of the moveable carriage assembly. When this lever is pulled forward, the paper can be straightened or removed.

Line Spacer

To the right of the paper release lever is a lever which permits selection of single, one and one-half, or double spacing between lines.

Platen

The roller, around which paper is inserted and against which the type bars strike, is called the platen.

Carriage Release (right and left)

There are two Carriage Release buttons, one located to the right and one to the left of the platen. When either or both are held down, the entire carriage assembly can be freely moved right or left. The carriage should not be moved when power is off.

Platen Knobs (right and left)

The platen knobs, located at each end of the platen, are used for turning the platen forward or backward.

Platen Variable

When this button, located in the center of the left platen knob, is depressed, the platen is released to allow the operator to position the paper at other than standard line spacing. Releasing the button restores standard line spacing.

Margin Release

This lever, which is located behind the left platen knob, can be raised to move the carriage to the left of the margin stop.

Ribbon Positioner

The ribbon position lever, located on the right side of the typewriter below the carriage, positions the ribbon for typing through its upper or lower part, or for typing stencils.

COND STOP (')

This key is used to punch a Conditional Stop code ('') into paper tape. When sensed by the tape reader, this code stops the reader and sends a start signal

to the computer.

TAB

This key moves the carriage to the next established tab position.

COLOR SHIFT

This key shifts and locks the ribbon for typing through its upper or lower half.

CAR RET

This key returns the carriage to the left margin and spaces the paper to the next typing line.

BACK SPACE

This key moves the carriage back one letter space.

Space

This bar moves the carriage forward one letter space.

LOWER CASE, UPPER CASE

These keys lock the keyboard in position for typing lower or upper case characters.

TAPE INTERLOCK

Both the reader and punch contain a tape interlock that stops the device if the tape breaks or if the supply is exhausted.

FEED KNOBS (Reader and Punch)

The reader and punch feed knobs are located on the left of the read and punch heads, respectively. These knobs can be used to move the tape forward or backward manually.

OPTIONAL EQUIPMENT

Model 101 Visual Oscilloscope Display

The Visual Oscilloscope Display is a separate unit consisting of an oscilloscope and the controls necessary to adjust the images displayed on it.

The oscilloscope furnishes the operator with a visual representation in binary form of the content of the three control registers: Counter, Instruction, and Accumulator.

The Counter Register displays the address of the next instruction to be executed and an indication if overflow has occurred since the last overflow condition interrogation. A "1" in the sign position of this register indicates that overflow has occurred; a "0" that it has not.

The Instruction Register displays the last instruction executed, or the second operand, in the case of multiplication and division.

The Accumulator displays the results of the last computation or data manipulation. See Figure 15.

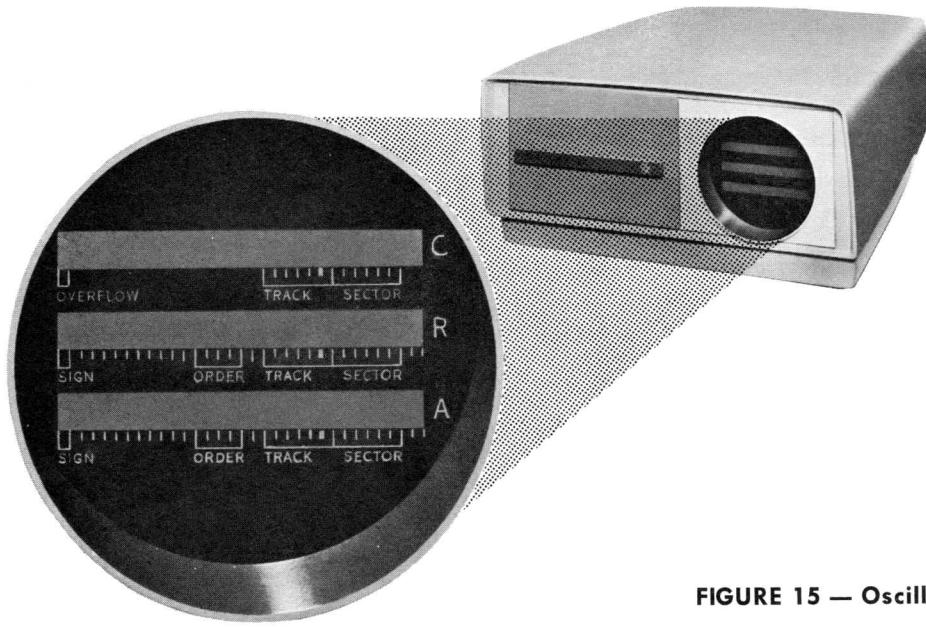


FIGURE 15 — Oscilloscope

On the left side of the oscilloscope are controls for adjusting the images of the respective control registers (Figure 16).

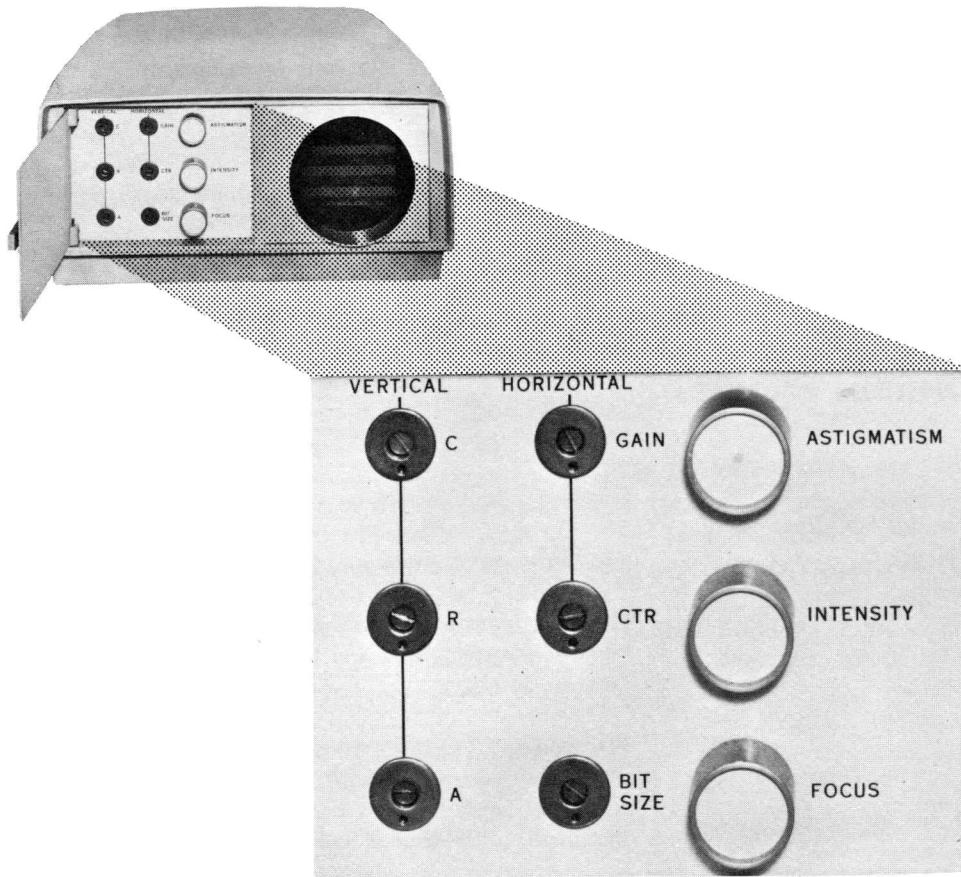


FIGURE 16 — Oscilloscope Controls

Adjustments are possible for each register by turning the appropriate screw under the designations HORIZONTAL or VERTICAL. The screws are labeled C for Counter Register, R for Instruction Register, and A for Accumulator. By turning a screw under VERTICAL, the operator can adjust the vertical dis-

placement of the designated register. Beside each of the screws in the HORIZONTAL row are labels specifying the kind of adjustment that can be effected by turning the particular screw: GAIN increases or decreases the width of the bit patterns, CTR centers the display in the windows, and BIT SIZE increases or decreases the amplitude of the bits displayed. Each of these screws affects all three register displays. The three knobs are also labeled to indicate their functions: ASTIGMATISM, INTENSITY, and FOCUS. ASTIGMATISM and FOCUS are used together to adjust the sharpness of the trace, and INTENSITY is used to make the image more or less brilliant.

Model 141 Paper-Tape Reader

The Model 141 Paper-Tape Reader is a self-contained, asynchronous device for reading any standard 5 to 8 channel perforated tape. Its reading speed is rated up to 60 characters-per-second. Tape can move forward or in reverse. Tape wear is exceptionally low, and standard paper-tapes may be used hundreds of times. See Figure 17.

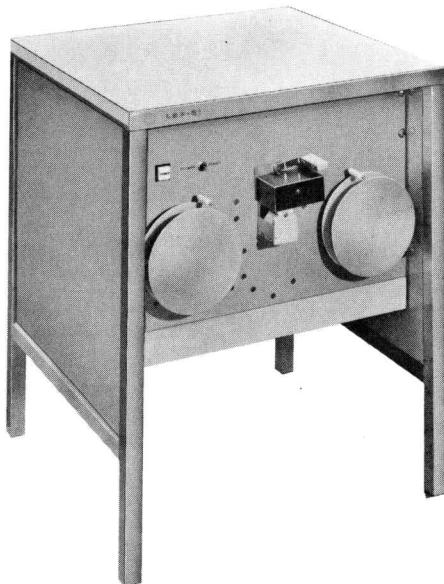


FIGURE 17 — Model 141 Paper-Tape Reader

The tape transporting equipment consists of two reels, mounted one on each side of the read head. They are rotated by a gear and brake mechanism actuated by tape tension arms. Either reel can be the take-up or supply reel, depending upon the direction of tape movement.

The Reader is controlled by two switches: a button-type POWER switch and a toggle switch for positioning the tape. The unit may be turned ON by depressing the POWER switch. Depressing the switch a second time turns the unit OFF. This switch is lighted while the unit is ON. The toggle switch is a momentary one which controls the bi-directional movement of the tape. By holding this switch in the direction the tape is to move, the operator can position a tape in the read head. Information enters the Accumulator of the computer only when the tape is moving forward (i. e., right to left). Tapes may be rewound at speeds up to 300 characters-per-second.

Model 151 Paper-Tape Punch

The Model 151 Paper-Tape Punch is a self-contained asynchronous unit capable of perforating paper-tape at speeds up to 60 characters-per-second. It is equipped with two reels: a supply reel, mounted behind the right side of the panel,

and a take-up reel, mounted on the left front-side of the panel. The take-up reel is rotated by a gear and brake mechanism activated by the tape tension arm. See Figure 18.

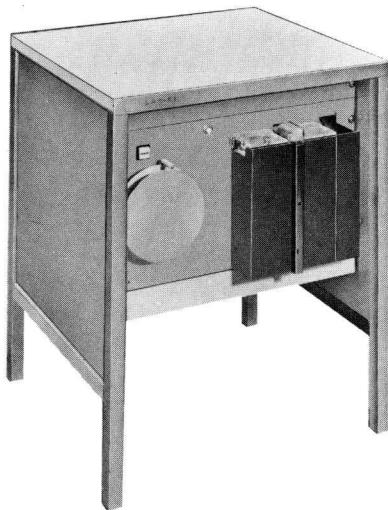


FIGURE 18 — Model 151 Paper-Tape Punch

The punch is controlled by two button-type switches: a POWER switch and a tape feed switch. The POWER switch is a momentary switch which is depressed to turn the power ON; a second depression turns it OFF. The switch is lighted when the unit is ON. The tape feed switch is also a momentary switch. When it is depressed and held, tape is fed to the punch head which punches sprocket (feed) holes only.

PROGRAM AND DATA MANIPULATION

PROGRAM INPUT ROUTINE

Since all computing and data handling in the LGP-21 is done with binary numbers it is necessary to have a program that will read decimaly coded programs, convert them to binary, and store them in designated locations. Such a program is called a program input routine and is provided to all users. It also contains several utility features which are useful in checking out a program.

Bootstrap

In order to load the program input routine itself into the computer, a series of starting instructions must be stored in memory manually. This group of instructions is called the bootstrap, and the process of loading it is called the bootstrap procedure. The bootstrap consists of three instructions stored in memory and a fourth instruction that is placed in the Instruction Register and executed. When executed, this fourth instruction transfers control to the bootstrap allowing it to read in a "hexadecimal fill" sequence which in turn loads the program input routine.

Bootstrap Procedure

The following sequence of operations will manually input the four instructions necessary to initiate the automatic loading of the hexadecimal fill sequence and subsequent program input routine.

1. Depress STOP READ on the typewriter.
2. Place bootstrap tape in reader.
3. Depress I/O switch on console.
4. See that all typewriter levers are UP.
5. Set MODE switch on console to MAN INPUT.
6. Depress START READ on typewriter.
7. Depress FILL CLEAR switch on console.
8. Depress START READ on typewriter.
9. Set MODE switch on console to ONE OPER.
10. Depress EXECUTE switch on console.
11. Repeat steps 5 through 10 twice.
12. Set MODE switch on console to MAN INPUT.
13. Depress START READ on typewriter.

14. Depress FILL CLEAR switch on console.
15. Set MODE switch on console to ONE OPER.
16. Depress EXECUTE switch on console.
17. Set MODE switch on console to NORMAL.
18. Depress START switch on console.

SUBROUTINES

A subroutine is a set of instructions that performs a standard task which is of sufficiently general nature to be used by a number of different programs. Examples are subroutines to input and output data, to compute a square root, to invert a matrix, etc. A comprehensive library of general purpose, mathematical, statistical, scientific, engineering, and utility routines is available to all LGP-21 users.

Calling Sequence

In order for a subroutine to function, computer control must be transferred from a main program to the subroutine. Also, any information required by the subroutine for its operation (such as the argument for which a square root is to be computed, or the locations where data are to be stored) must be supplied to it. Then, when the subroutine has finished its operations, control must be returned to the main program. The instructions in the main program which provide the information required by the subroutine and which transfer control to it are referred to as a calling sequence.

The Set Return Address (R) instruction is normally used to set the exit of the subroutine so it will return control to the appropriate instruction in the main program. The next instruction after the R will normally be an Unconditional Transfer (U) to the entry location of the subroutine. Following the U instruction may be code words for the subroutine's use, or the next instruction to be executed after the subroutine. Some subroutines require that specific information be in the Accumulator when the R/U instructions are executed. In any case, the written description of a subroutine always specifies how the calling sequence is to be written.

TAPE CODES

The punched paper-tape contains 6 channels numbered as shown in Figure 19.

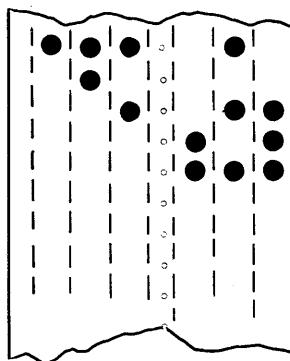
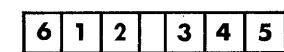


FIGURE 19 — Tape Channels

Any or all of these channels may have a punch in one horizontal row. A combination of punches in channels 1 through 6 is called a character and represents the bits which will enter the computer. Bits 5 and 6 are called zone bits since they differentiate numeric, alphabetic, and typewriter control characters for the computer. Table III, Input and Output Codes, lists the binary code equivalents of all typewriter characters and controls.

DATA INPUT

Characters read from tape or typed on the keyboard can enter the Accumulator 4 bits or 6 bits at a time. Depending upon whether 4-bit or 6-bit input is specified, characters enter the Accumulator as channels 1, 2, 3, 4 in bit positions 28 through 31, or as channels 1, 2, 3, 4, 5, 6 in bit positions 26 through 31, entering at the right and moving left. Eight 4-bit characters fill 32 bits, and five 6-bit characters fill 30 bits of the Accumulator. Entering additional characters causes the characters on the left to be lost. When input through the typewriter, all codes except Delete, Tape Feed and Conditional Stop enter in 6-bit; no control codes, except Space, enter in 4-bit.

In the Normal mode of operation, the input device must be activated by the execution of an input instruction. The input instruction stops the computer and starts the input device. The track-address of the input instruction selects the input device, and the sign determines whether input will be 4-bit or 6-bit. When a Conditional Stop code (!) is read on tape or when the START COMP lever on the typewriter is depressed during manual input, the input device is made inactive, and the computer proceeds to the next instruction. The 141 Reader will also stop when a stop code is pushed past bit position zero of the Accumulator. In Manual Input mode of operation, 4-bit input from the typewriter is automatically selected.

DATA OUTPUT

Information can be transmitted, one character at a time, from the computer to a punch or to the typewriter. Depending upon whether 4-bit or 6-bit output is selected, the character code represented by bits 0 through 3 of the Accumulator with zone bits 1, 0, or the character code represented by bits 0 through 5 of the Accumulator, will be transmitted. The contents of the Accumulator will be unchanged.

The output device must be activated by the execution of an output instruction. The track-address of the output instruction selects the output device, and the sign determines whether output will be 4-bit or 6-bit. After the output action is initiated, the LGP-21 does not stop but may continue computing. If a second output instruction is encountered before the first output operation is completed, and both instructions selected the same output unit, the computer will wait until the first one is completed before initiating the second. It will execute the second instruction without waiting for the first to be completed, if two different output units are selected.

Figure 20 shows the track addresses to be used with the Input and Print commands for selection of the various devices.

<u>Track-address</u>	<u>Device</u>
00	Model 141 Paper-Tape Reader
06	Model 151 Paper-Tape Punch
02	Tape Typewriter
62 <i>(hexadecimal 3Q)</i>	Input shift (does not select an input device).

FIGURE 20 — Input/Output Assignments

SCALING

The binary point indicates the separation between the integral and fractional portions of a binary number. The binary point in the LGP-21 Computer is always after the sign and before bit position 1, so the computer considers every number as less than +1. However, the programmer can assume the binary point to be wherever he wants. Positioning the assumed binary point is called scaling. The number of bits away from the computer point that the assumed point is placed is called the scale factor, or "q" of the number. The purpose of scaling is to shift data into favorable positions to eliminate overflow or loss of precision, and to effect binary integers.

Figure 21 shows how scaling affects the positioning of the number in memory.

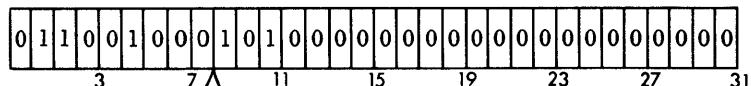


FIGURE 21 — 200.625 @ q = 8

The q in Arithmetic Operations

The binary points of the operands in addition or subtraction must be aligned, and the results of an add or a subtract will be at the "q" of the operands. The "q's" of the operands in multiplication and division need not be the same. In multiplication, the "q" of the multiplier plus the "q" of the multiplicand equals the "q" of the product. In division, the "q" of the dividend minus the "q" of the divisor equals the "q" of the quotient.

Following a multiply operation, a 62-bit product is developed. In the double-length product, each bit in the least significant half (31-62) or the most significant half (0-31) is 31 positions away from the corresponding bit position in the other half; for example, bits 3 and 34, bits 12 and 43. With the N-multiply, bits 31-62 are held in the Accumulator and become bits 0-30 with a spacer bit of zero. The "q" of the product of an N-multiply can be figured as: The "q" of the multiplier plus the "q" of the multiplicand less 31 equals the "q" of the result.

For a representation of the extended Accumulator, see Figure 22.

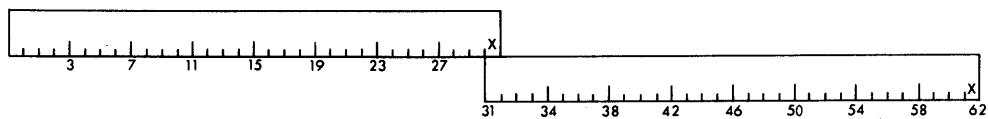


FIGURE 22 — Extended Accumulator

SHIFTING

In addition to the 4- and 6-bit shifts which can be accomplished with an I command, shifts may also be achieved through multiplication and division:

Any number M-multiplied by $1 @ q = n$ shifts the number in the Accumulator n bit-positions to the right.

Dividing by $1 @ q = n$ shifts the contents of the Accumulator n bit-positions to the left. It is possible to cause overflow when shifting by means of a divide.

N-multiplying by $1 @ q = n$ shifts the number in the Accumulator left $31 - n$ places without the possibility of overflow although the number may be truncated on the left.

NEGATIVE NUMBERS

Except for N-multiply, when an arithmetic operation is performed, the computer considers the signs of both values and determines the sign of the result in accordance with algebraic rules. In addition to having a "1" in the sign position, a negative number is carried as the one's complement plus $1 @ q = 30$ (which is the two's complement).

0000000000000000000000000000000010111000 = 5.75 @ q = 26

1111111111111111111111111111101000110 = one's complement of 5.75 @ q = 26

1111111111111111111111111111101001000 = -5.75 @ q = 26 = two's complement

To determine the bit configuration of a negative number, write it as a positive number at the appropriate "q" and then reverse all bits to the left of the least significant bit. The least significant bit and the zero bits to the right of it are unchanged.

HEXADECIMAL NOTATION

Since it is inconvenient to write 32 bits each time a binary value is to be expressed, it becomes apparent that another form of notation is needed. Any numeric digit can be represented in four binary bits. This uses ten of 16 possible combinations of four bits. The remaining six combinations are the binary values for decimal 10, 11, 12, 13, 14, and 15. The LGP-21 gives these values the alphabetic designations F, G, J, K, Q, and W, respectively. The number system using 0 through 9 and F, G, J, K, Q, W is a system to the base 16 and is called hexadecimal.

Hexadecimal notation is commonly used to describe binary words.

0000 1011 0011 1010 0011 0101 1111 0000
0 G 3 F 3 5 W 0

This discussion concludes the preliminary version of the LGP-21 Reference Manual. As a final aid to new users of the LGP-21, a number of reference tables are attached. Table I provides a list of decimal and hexadecimal equivalents of commands and addresses; Table II gives the Powers of Two; and Table III provides a complete listing of input and output codes.

For a more complete discussion of programming techniques and procedures for the LGP-21, the reader is referred to the LGP-21 Programming Manual.

DECIMAL AND HEXADECIMAL EQUIVALENTS OF COMMANDS AND ADDRESSES

TABLE I

<u>Symbol</u>	<u>Command</u>	<u>Binary</u>	<u>Hexadecimal</u>	<u>Decimal</u>
Z	Stop; Sense and Transfer	0000	0	0
B	Bring	0001	1	1
Y	Store Address	0010	2	2
R	Set Return Address	0011	3	3
I	Input; Input Shift	0100	4	4
D	Divide	0101	5	5
N	Multiply, Save Right	0110	6	6
M	Multiply, Save Left	0111	7	7
P	Print or Punch	1000	8	8
E	Extract	1001	9	9
U	Unconditional Transfer	1010	F	10
T	Conditional Transfer	1011	G	11
H	Hold	1100	J	12
C	Clear	1101	K	13
A	Add	1110	Q	14
S	Subtract	1111	W	15

Decimal	Track Hexadecimal	Sector Hexadecimal	Decimal	Track Hexadecimal	Sector Hexadecimal
0	00	00	32	20	80
1	01	04	33	21	84
2	02	08	34	22	88
3	03	0j	35	23	8j
4	04	10	36	24	90
5	05	14	37	25	94
6	06	18	38	26	98
7	07	1j	39	27	9j
8	08	20	40	28	f0
9	09	24	41	29	f4
10	0f	28	42	2f	f8
11	0g	2j	43	2g	fj
12	0j	30	44	2j	g0
13	0k	34	45	2k	g4
14	0q	38	46	2q	g8
15	0w	3j	47	2w	gj
16	10	40	48	30	j0
17	11	44	49	31	j4
18	12	48	50	32	j8
19	13	4j	51	33	jj
20	14	50	52	34	k0
21	15	54	53	35	k4
22	16	58	54	36	k8
23	17	5j	55	37	kj
24	18	60	56	38	q0
25	19	64	57	39	q4
26	1f	68	58	3f	q8
27	1g	6j	59	3g	qj
28	1j	70	60	3j	w0
29	1k	74	61	3k	w4
30	1q	78	62	3q	w8
31	1w	7j	63	3w	wj

POWERS OF 2 TABLE

TABLE II

2^n	n	2^{-n}
1	0	1. 0
2	1	0. 5
4	2	0. 25
8	3	0. 125
16	4	0. 062 5
32	5	0. 031 25
64	6	0. 015 625
128	7	0. 007 812 5
256	8	0. 003 906 25
512	9	0. 001 953 125
1 024	10	0. 000 976 562 5
2 048	11	0. 000 488 281 25
4 096	12	0. 000 244 140 625
8 192	13	0. 000 122 070 312 5
16 384	14	0. 000 061 035 156 25
32 768	15	0. 000 030 517 578 125
65 536	16	0. 000 015 258 789 062 5
131 072	17	0. 000 007 629 394 531 25
262 144	18	0. 000 003 814 697 265 625
524 288	19	0. 000 001 907 348 632 812 5
1 048 576	20	0. 000 000 953 674 316 406 25
2 097 152	21	0. 000 000 476 837 158 203 125
4 194 304	22	0. 000 000 238 418 579 101 562 5
8 388 608	23	0. 000 000 119 209 289 550 781 25
16 777 216	24	0. 000 000 059 604 644 775 390 625
33 554 432	25	0. 000 000 029 802 322 387 695 312 5
67 108 864	26	0. 000 000 014 901 161 193 847 656 25
134 217 728	27	0. 000 000 007 450 580 596 923 828 125
268 435 456	28	0. 000 000 003 725 290 298 461 914 062 5
536 870 912	29	0. 000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0. 000 000 000 931 322 574 615 478 515 625
2 147 483 648	31	0. 000 000 000 465 661 287 307 739 257 812 5

INPUT/OUTPUT CODES

TABLE III

Character Codes	Hexadecimal Accumulator Value	Tape Codes	Input Codes			Output Codes
			6	1234	56	
) 0	0800 0000	0 0000 1	0000	10		0000 10
L 1	1800 0000	0 0001 1	0001	10		0001 10
* 2	2800 0000	0 0010 1	0010	10		0010 10
" 3	3800 0000	0 0011 1	0011	10		0011 10
Δ 4	4800 0000	0 0100 1	0100	10		0100 10
% 5	5800 0000	0 0101 1	0101	10		0101 10
\$ 6	6800 0000	0 0110 1	0110	10		0110 10
π 7	7800 0000	0 0111 1	0111	10		0111 10
Σ 8	8800 0000	0 1000 1	1000	10		1000 10
(9	9800 0000	0 1001 1	1001	10		1001 10
F f	f800 0000	0 1010 1	1010	10		1010 10
G g	g800 0000	0 1011 1	1011	10		1011 10
J j	j800 0000	0 1100 1	1100	10		1100 10
K k	k800 0000	0 1101 1	1101	10		1101 10
Q q	q800 0000	0 1110 1	1110	10		1110 10
W w	w800 0000	0 1111 1	1111	10		1111 10
Z z	z400 0000	1 0000 0	0000	01		0000 01
B b	b400 0000	1 0001 0	0001	01		0001 01
Y y	y400 0000	1 0010 0	0010	01		0010 01
R r	r400 0000	1 0011 0	0011	01		0011 01
I i	i400 0000	1 0100 0	0100	01		0100 01
D d	d400 0000	1 0101 0	0101	01		0101 01
N n	n400 0000	1 0110 0	0110	01		0110 01
M m	m400 0000	1 0111 0	0111	01		0111 01
P p	p400 0000	1 1000 0	1000	01		1000 01
E e	e400 0000	1 1001 0	1001	01		1001 01
U u	u400 0000	1 1010 0	1010	01		1010 01
T t	t400 0000	1 1011 0	1011	01		1011 01
H h	h400 0000	1 1100 0	1100	01		1100 01
C c	c400 0000	1 1101 0	1101	01		1101 01
A a	a400 0000	1 1110 0	1110	01		1110 01
S s	s400 0000	1 1111 0	1111	01		1111 01
Lower Case	1000 0000	0 0001 0	*0001	00		0001 00
Upper Case	2000 0000	0 0010 0	*0010	00		0010 00
Color Shift	3000 0000	0 0011 0	*0011	00		0011 00
Car. Return	4000 0000	0 0100 0	*0100	00		0100 00
Back Space	5000 0000	0 0101 0	*0101	00		0101 00
Tab	6000 0000	0 0110 0	*0110	00		0110 00
Cond. Stop	8000 0000	0 1000 0				1000 00
Space	0j00 0000	1 0000 1	0000	11		0000 11
-	1j00 0000	1 0001 1	0001	11		0001 11
= +	2j00 0000	1 0010 1	0010	11		0010 11
:	3j00 0000	1 0011 1	0011	11		0011 11
?	/ 4j00 0000	1 0100 1	0100	11		0100 11
]	. 5j00 0000	1 0101 1	0101	11		0101 11
[,	6j00 0000	1 0110 1	0110	11		0110 11
V v	7j00 0000	1 0111 1	0111	11		0111 11
O o	8j00 0000	1 1000 1	1000	11		1000 11
X x	9j00 0000	1 1001 1	1001	11		1001 11
Delete	wj00 0000	1 1111 1				1111 11

* 6-bit input only

COMMERCIAL COMPUTER DIVISION

 **GENERAL
PRECISION**

INFORMATION SYSTEMS GROUP